

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-031660

(43)Date of publication of application : 03.02.1998

---

(51)Int.Cl. G06F 17/00

G06F 3/14

G06F 12/00

---

(21)Application number : 08-302511 (71)Applicant : FUJI XEROX CO LTD

(22)Date of filing : 28.10.1996 (72)Inventor : NOMURA YASUHIKO

HAYASHI KOICHI

---

(30)Priority

Priority number : 08148366

Priority date : 17.05.1996

Priority country : JP

---

(54) INFORMATION PROCESSOR

(57)Abstract:

PROBLEM TO BE SOLVED: To provide an information processor with which information required for activity is put in order for each job by automatically constructing the environment of working for holding the information while being reflected with the activity performed in the working environment.

SOLUTION: The working environment is defined as an information unit cluster (snap) and a data object or the like to be utilized in the working environment is defined as an information unit (meditator). Then, the plural information unit

clusters are stored in an information unit cluster holding history storage part 3 while collected for each place and on the other hand, the information unit cluster as a working object is held in an information unit cluster holding part 14. When prescribed conditions such as the change of job are arranged, the information unit cluster held in the information unit cluster holding part 14 is stored in the information unit cluster holding history storage part 3 by a storage instructing part 7, and selection information such as a job name is applied to this information unit cluster by a selection information applying part 6. When the selection information is inputted from a user in order to define a certain information unit cluster as the working object, while using this information, an information unit cluster selecting part 4 selects the relevant information unit cluster from the storage part 3 and it is read out to the information unit cluster holding part 14 by an information unit cluster reading part 5.

---

LEGAL STATUS [Date of request for examination] 11.09.1998

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of

rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3279201

[Date of registration] 22.02.2002

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

**\* NOTICES \***

**JPO and INPIT are not responsible for any**

**damages caused by the use of this translation.**

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.\*\*\*\* shows the word which can not be translated.

3.In the drawings, any words are not translated.

---

**CLAIMS**

---

[Claim(s)]

[Claim 1] It is constituted including the application object for operating a data



object and said data object. In an information processor equipped with the workspace for collecting and creating said data object using said application object An information-unit set maintenance means to hold the set of the information unit which is a data object or an application object, The information-unit set maintenance hysteresis storage means which can memorize two or more said information-unit sets, A storage directions means to make said information-unit set maintenance hysteresis storage means memorize said information-unit set which said information-unit set maintenance means holds, A selection information grant means to give selection information to said information-unit set memorized by said storage directions means to said information-unit set maintenance hysteresis storage means, A selection means to choose from from among said information-unit sets memorized by said information-unit set maintenance hysteresis storage means using said selection information to which said information-unit set which said information-unit set maintenance means is made to memorize was given by said selection information grant means, The information processor characterized by having the read-out means which reads said information-unit set chosen by said selection means to said information-unit set maintenance means.

[Claim 2] An information-unit edit means to edit an information unit in an information processor according to claim 1 to said information-unit set which said

information-unit set maintenance means holds, An increase-and-decrease name storage means of an information unit to memorize said information-unit name added or deleted by said information-unit edit means, Furthermore, have and said selection information grant means is given to said information-unit set by making into selection information said information-unit name which said increase-and-decrease name storage means of an information unit memorizes as said selection information. Said selection means is an information processor characterized by choosing said information-unit set using said information-unit name given by said selection information grant means.

[Claim 3] An information-unit directions means to direct the information unit which said information-unit set maintenance means holds in an information processor according to claim 1 or 2, The 2nd selection means which chooses said information-unit set within said information-unit set maintenance hysteresis storage means used as the point which moves or copies said information unit directed by said information-unit directions means, An information-unit migration means to add said information unit directed by said information-unit directions means to said information-unit set chosen by said 2nd selection means, A moved material information-unit set name storage means to memorize the each set name of said information unit of a migration place a moved material based on being between information-unit sets and moving said information unit with said

information-unit migration means, Furthermore, have and said selection information grant means stores the information-unit set name of the migration origin which said moved material information-unit set name storage means memorizes as said selection information. Said selection means is an information processor characterized by choosing said information-unit set using said moved material information-unit set name given by said selection information grant means.

[Claim 4] In an information processor given in any 1 term of claim 1 thru/or claim

3 An information-unit directions means to receive the directions selection made into a processing object to said information unit which said information-unit set maintenance means holds, An information-unit processing means to process to the information unit directed by said information-unit directions means, An execution information unit name storage means to memorize the information-unit name by which processing was performed by said information-unit processing means, Furthermore, have and said selection information grant means is given to said information-unit set by making into selection information said information-unit name which said execution information unit name storage means memorizes as said selection information. Said selection means is an information processor characterized by choosing said information-unit set using said information-unit name given by said selection

information grant means.

[Claim 5] It has further an execute command name storage means to memorize the command name performed to said information unit by said information-unit processing means in an information processor according to claim 4. Said selection information grant means is given to said information-unit set by making into selection information said command name which said execute command name storage means memorizes as said selection information. Said selection means The information processor characterized by choosing said information-unit set using said command name given by said selection information grant means.

[Claim 6] In an information processor given in any 1 term of claim 1 thru/or claim 5 The selection information classification display means which indicates by classification said selection information given to said information-unit set memorized by said information-unit set maintenance hysteresis storage means, A selection information sorting-page-format maintenance means to hold the display format of the selection information displayed by said selection information classification display means, A selection information directions means to direct said selection information displayed by said selection information classification display means, Furthermore, have, and said information-unit set maintenance hysteresis storage means divides said

selection information for every one or more selection information attributes, and give it to said information-unit set and it memorizes it. Said selection information classification display means has the 1st hierarchy's selection information attribute viewing area, and the 2nd hierarchy's selection information attribute viewing area. The selection information of said selection information attribute is shown a list table in the format set to said 1st hierarchy's selection information attribute viewing area by said selection information sorting-page-format maintenance means as the 1st hierarchy. The selection information of said selection information attribute is shown a list table in the format set to said 2nd hierarchy's selection information attribute viewing area by said selection information sorting-page-format maintenance means as the 2nd hierarchy. With said selection information directions means Only the 2nd hierarchy's selection information given to the information-unit set memorized by said information-unit set maintenance hysteresis storage means which contains the directed selection information concerned based on said selection information of said 1st hierarchy having been directed is displayed on said 2nd hierarchy's selection information attribute viewing area. Said selection means is an information processor characterized by choosing one of said the information-unit sets which said information-unit set maintenance hysteresis storage means memorizes using the both sides of the selection information directed on said one hierarchy, and

the selection information directed on said 2nd hierarchy.

[Claim 7] It is the information processor which is further equipped with a selection-information newest time maintenance means hold the date and time of creation of said information-unit set of the newest to which each selection information is given to all said selection information given to said information-unit set which said information-unit set maintenance hysteresis storage means memorizes in the information processor according to claim 6, and is characterized by to rearrange said selection-information classification display means by said newest date and time of creation to which said selection-information newest time maintenance means holds said selection information, and to display it.

[Claim 8] It has further a selection information grant time maintenance means to hold the time given corresponding to the date and time of creation of all the information-unit sets of said to said all selection information given to said information-unit set which said information-unit set maintenance hysteresis storage means memorizes in the information processor according to claim 6. Said selection information classification display means is an information processor characterized by rearranging and displaying said selection information corresponding to the time which said selection information grant time maintenance means holds.

[Claim 9] In an information processor given in any 1 term of claim 6 thru/or claim 8 Said selection information classification display means displays said selection information given to said information-unit set memorized by said information-unit set maintenance hysteresis storage means. Said selection means with said selection information classification display means The information processor characterized by choosing the newest thing in the information-unit set which has the selection information which received directions of the selection information given to said information-unit set read into said information-unit set maintenance means of said displayed selection information, and was directed.

[Claim 10] In an information processor given in any 1 term of claim 6 thru/or claim 8 It has further a selection information input means to make the selection information given to said information-unit set memorized for said information-unit set maintenance hysteresis storage means based on directing storage of said information-unit set for said information-unit set maintenance hysteresis storage means with said storage directions means input or direct by the user. Said selection information input means receives the input of a work name which the user was performing at the time of storing of said information-unit set. Said selection information classification display means is an information processor characterized by indicating said information-unit set for said work name memorized for said information-unit set maintenance hysteresis storage means

by classification by a display or said work name.

[Claim 11] It is the information processor characterized by indicating said information-unit set for the user name which memorizes said selection information classification display means for said information-unit set maintenance hysteresis storage means in an information processor according to claim 6 by classification by a display or the user name.

[Claim 12] It is the information processor characterized by indicating the information-unit set name for said information-unit name which memorizes said selection information classification display means for said information-unit set maintenance hysteresis storage means in an information processor according to claim 6 by classification by a display or said information-unit name.

[Claim 13] In an information processor given in any 1 term of claim 1 thru/or claim 12 The 3rd selection means which chooses said two or more [ which was held at said information-unit set maintenance means or said information-unit set maintenance hysteresis storage means ] information-unit sets, It has further an information-unit set composition means to perform the set operation of said information unit which two or more information-unit sets chosen by said 3rd selection means hold. The information processor characterized by considering as said information-unit set which reads into said information-unit set maintenance means the new information-unit set created by performing said set



operation, and said information-unit set maintenance means holds.

[Claim 14] The 4th selection means which chooses said information-unit set held at said information-unit set maintenance means or said information-unit set maintenance hysteresis storage means in an information processor according to claim 13, A narrowing-down condition directions means to direct the narrowing-down conditions of an information unit to the set of the information unit which the information-unit set chosen by said 4th selection means holds, A narrowing-down activation means to delete the thing which was directed by said narrowing-down condition directions means and which narrows down and does not fulfill conditions from said information-unit set chosen by said 4th selection means, Furthermore, the information processor characterized by considering as the information-unit set which has, narrows down said information unit to said information-unit set, creates a new information-unit set, and reads the new information-unit set concerned into said information-unit set maintenance means, and said information-unit set maintenance means holds.

[Claim 15] In an information processor given in any 1 term of claim 1 thru/or claim 14 It has further a storage activation decision means to judge whether an information-unit set is stored in said information-unit set maintenance hysteresis storage means based on the user or the event by the directions from a command having occurred based on the Ruhr which was able to be appointed beforehand.

Said information-unit edit means is an information processor characterized by storing in said information-unit set maintenance hysteresis storage means the information-unit set which said information-unit set maintenance means holds based on decision by said storage activation decision means.

[Claim 16] It is the information processor a storage activation decision means carries out said decision in an information processor according to claim 15 based on the event which reads the information-unit set which said information-unit set maintenance hysteresis storage means memorizes to said information-unit set maintenance means having generated, and carry out that said information-unit edit means stores the information-unit set said information-unit set maintenance means holds at the time of the event generating concerned to said information-unit set maintenance hysteresis storage means as the description.

[Claim 17] It is the information processor have further a change manifestation means receive manifestation of change of the class of work from a user, in an information processor according to claim 15, and said storage activation decision means carries out said decision based on the manifestation event from said change manifestation means having generated, and carry out that said information-unit edit means stores the information-unit set said information-unit set maintenance means holds at the time of the event generating concerned to

said information-unit set maintenance hysteresis storage means as the description.

[Claim 18] It is the information processor which it has further the user management tool which distinguishes a user uniquely in an information processor according to claim 15, and said storage activation decision means makes said judgment based on the event from which said user management tool distinguished change of a user having occurred, and carries out [ that said information-unit edit means stores the information-unit set which said information-unit set maintenance means holds at the time of the event generating concerned to said information-unit set maintenance hysteresis storage means, and ] as the description.

[Claim 19] It has further an information-unit modification detection means to detect the addition, deletion, or attribute modification of the information unit of an information-unit set which said information-unit set maintenance means holds in an information processor according to claim 15. Said storage activation decision means makes said judgment based on the event in which said information-unit modification detection means detected said modification having occurred. Said information-unit edit means is an information processor characterized by storing in said information-unit set maintenance hysteresis storage means the information-unit set which said information-unit set maintenance means holds at

the time of the event generating concerned.

[Claim 20] It has further an information-unit display detection means to detect having displayed the contents of the information unit used as a processing object in the information processor according to claim 15. Said storage activation decision means makes said judgment based on the event in which said information-unit display detection means detected said display having occurred. Said information-unit edit means is an information processor characterized by storing in said information-unit set maintenance hysteresis storage means the information-unit set which said information-unit set maintenance means holds at the time of the event generating concerned.

[Claim 21] It is the information processor have further a storage execution cycle maintenance means hold the period which makes said information-unit set maintenance hysteresis storage means memorize an information-unit set in an information processor according to claim 15, and said storage activation decision means carries out said decision based on the event for every period of said, and carry out that said information-unit edit means stores the information-unit set said information-unit set maintenance means holds at the time of the event generating concerned to said information-unit set maintenance hysteresis storage means as the description.

[Claim 22] A storage activation regulation maintenance means to hold the

regulation based on the count by which the command was executed in an information processor according to claim 15, It has further a command storage means to memorize one or more executed commands. Said storage activation decision means makes said judgment based on the event whose command defined beforehand corresponded with the sequence of the regulation which said storage activation regulation maintenance means holds. Said information-unit edit means is an information processor characterized by storing in said information-unit set maintenance hysteresis storage means the information-unit set which said information-unit set maintenance means holds at the time of the event generating concerned.

[Claim 23] It has further an information-unit migration detection means to detect that the information unit was moved to said information-unit set maintenance hysteresis storage means from said information-unit set maintenance means by said information-unit migration means in an information processor according to claim 15. Said storage activation decision means makes said judgment based on the event in which said information-unit migration detection means detected said migration having occurred. Said information-unit edit means is an information processor characterized by storing in said information-unit set maintenance hysteresis storage means the information-unit set which said information-unit set maintenance means holds at the time of the event

generating

concerned.

[Claim 24] In an information processor according to claim 1 said information-unit set maintenance means A current condition maintenance means to hold two or more information-unit sets used as the candidate for a display with the selection information as the attribute, The current status-display means which carries out the display output of the information-unit set corresponding to the selection information inputted by the user of the information-unit sets held at the current condition maintenance means concerned, A preservation decision means to judge whether a preparation and said storage directions means save the information-unit set before the modification concerned when the number of the information units under information-unit set is changed according to the regulation set up beforehand, The information processor characterized by having an are recording means to make the information-unit set judged to save with the preservation decision means concerned memorize to said information-unit set maintenance hysteresis storage means.

[Claim 25] It is the information processor characterized by carrying out the display output of the information for identifying all information-unit sets said whose current condition maintenance means hold said current status-display means in an information processor according to claim 24.

[Claim 26] It is the information processor characterized by having further a

current condition acquisition means to make the place which had memorized two or more places where said information-unit set maintenance hysteresis storage means carried out grouping of two or more information-unit sets further in the information processor according to claim 24 or 25, and was chosen by the user of these places hold for said information-unit set maintenance means.

[Claim 27] It is the information processor characterized by having further a current condition acquisition means to make the place of either of these places hold for said information-unit set maintenance means based on the conditions which had memorized two or more places where said information-unit set maintenance hysteresis storage means carried out grouping of two or more information-unit sets further in the information processor according to claim 24 or 25, and set them up beforehand.

[Claim 28] It is the information processor characterized by judging it as what should be saved when the number of the information units under information-unit set changes said preservation decision means to reduction from an increment in an information processor according to claim 24.

[Claim 29] In an information processor according to claim 24 said selection means A related condition extract means to extract the information-unit set corresponding to the extract regulation about the selection information set up beforehand from among the information-unit sets held at said

information-unit set maintenance hysteresis storage means, A related condition maintenance means to hold the extracted information-unit set concerned, and list display means concerned to display the selection information of the extracted information-unit set to a user, A related information assignment means to specify the information-unit set corresponding to a demand of the user to the displayed selection information out of the information-unit set currently held at said related condition maintenance means, A preparation and the aforementioned read-out means are an information processor characterized by having the related condition installation means which reads the information-unit set specified by said related information assignment means from said related condition maintenance means, and is made to hold for said current condition maintenance means.

[Claim 30] In the information processor according to claim 29, said current condition maintenance means has the work name as selection information. Said related condition installation means Copy the information-unit set and work name which said specified information-unit set holds, and it is made to hold for said current condition maintenance means. Said related condition extract means The information processor characterized by extracting one or more [ which has the same work name as the work name of the information-unit set which said current condition maintenance means holds ] related information-unit sets.



[Claim 31] It is the information processor characterized by extracting the newest information-unit set for every user among the information-unit sets which other users who have the same work name as the work name which is the selection information of the information-unit set to which said present condition maintenance means holds said related condition extract means in an information processor according to claim 30 generated.

[Claim 32] It is the information processor characterized by extracting the newest information-unit set from said information-unit set maintenance hysteresis storage means for every work name which is the selection information among all information-unit sets whose users itself generated said related condition extract means by then in the information processor according to claim 30.

[Claim 33] It is the information processor which is further equipped with a condition composition means perform the set operation beforehand set up in the information processor according to claim 24 to the information unit which said current condition maintenance means holds, and which is the element of two or more information-unit sets, and is characterized by to add the condition composition means concerned to said current condition maintenance means, and to make the information-unit set compounded by performing the set operation concerned hold.

[Claim 34] It is the information processor carry out that the set operation of said

condition composition means takes the sum-set of the information unit under information-unit set which extracts the newest information-unit set for every user among the information-unit sets which other users who have the same work name as the work name from said information-unit set maintenance hysteresis storage means generated, and has the same work name concerned in an information processor according to claim 33 about the work name which is the selection information of the information-unit set which said present condition maintenance means holds as the description.

[Claim 35] In an information processor according to claim 33 the set operation of said condition composition means It is related with one work name which is the selection information of the information-unit set which said current condition maintenance means holds. Are newer than the information-unit set which a current condition maintenance means to have the same work name of the information-unit sets which other users who have the same work name generated holds. And the newest 1st information-unit set in an information-unit set of the user is extracted. Furthermore, the newest 2nd information-unit set in the information-unit set made before rather than the information-unit set which a present condition maintenance means to have the same work name among the information-unit sets which have the work name of the user holds is extracted. Take the difference of the information unit which said 2nd information-unit set

holds from the information unit which said 1st information-unit set holds, and the 3rd information-unit set is extracted. Furthermore, the information processor characterized by extracting the 4th information-unit set, adding the 4th information-unit set concerned to said present condition attaching part, and making it hold by taking the difference of the information unit which said 1st information-unit set holds from the information unit which the 3rd information-unit set holds.

[Claim 36] In an information processor according to claim 33 the set operation of said condition composition means It is related with one work name which is the selection information of the information-unit set which said current condition maintenance means holds. Are newer than the information-unit set which a current condition maintenance means to have the same work name of the information-unit sets which other users who have the same work name generated holds. And the newest 1st information-unit set in an information-unit set of the user is extracted. Furthermore, the newest 2nd information-unit set in the information-unit set made before rather than the information-unit set which a present condition maintenance means to have the same work name among the information-unit sets which have the work name of the user holds is extracted. Take the difference of the information unit which said 2nd information-unit set holds from the information unit which said 1st information-unit set holds about

each of two or more of other users, and each 3rd information-unit set of two or more of other users is extracted. Furthermore, take the sum-set of those the 3rd information-unit sets of all, and the 5th information-unit set is extracted. Furthermore, the information processor characterized by extracting the 4th information-unit set, adding the 4th information-unit set concerned to said present condition attaching part, and making it hold by taking the difference of the information unit which said 1st information-unit set holds from the information unit which the 5th information-unit set concerned holds.

[Claim 37] It is the information processor characterized by to take the common class of the information unit which the information-unit set which extracts the newest information-unit set for every user among the information-unit sets which other users who have the same work name as the work name specified as the selection information of the information-unit set to which said present condition maintenance means holds the set operation of said condition composition means in an information processor according to claim 33 generated, and has the same work name holds.

---

DETAILED DESCRIPTION

---

[Detailed Description of the Invention]

[0001]

[Field of the Invention] About the information processor which supports execution of business by offering the work environment (information-unit set) which holds objects (information unit), such as an accessible document, electronically, this invention relates to the information processor which offers the work environment which holds an object in connection with work, in order to perform two or more work in parallel especially.

[0002]

[Description of the Prior Art] Various office work can be carried out now by the end of today by performing file management with information management systems, such as a personal computer and a workstation. Resources, such as a document which these devices treat, an electronic mail, an application program, and an I/O device, are managed by the file system, a database, etc. It is effective to pack a required package of a document and a required instrument for every business in such management, when advancing business smoothly. The hierarchical directory and the thing to depend on a desktop are known by the conventional technique for packing the resource which can be treated with an information management system.

[0003] The hierarchical directory is adopted by the file system of an operating

system like UNIX or MS-DOS (trademark). A user can hold information collectively in the unit of a directory, and can constitute further two or more directories in a layered structure. In the information management system which adopted the hierarchical directory, a user can make a directory able to respond to business, can create, and can store the resource relevant to there. However, in such a conventional file system, each resource stored in the directory is shown to a user by the list of file names. For this reason, it was difficult to understand intuitively what kind of resource is used on business.

[0004] Then, the information management approach called a desktop was developed by Star (trade name) of Xerox Corp. A desktop holds individual work environment and offers the 2-dimensional flat surface expressing the desk top made into a metaphor. The resource relevant to business is arranged on a 2-dimensional flat surface as a graphic form called the icon corresponding to each class. Peripheral devices, such as a folder which packs data, such as a document, and two or more documents, and a printer, an application program, a reference, etc. are arranged as a symbol it is easier for a user to understand intuitively. In addition, in addition to this, instruments, such as a clock and a computer, and other resources are also arranged on a 2-dimensional flat surface. These icons can be freely arranged in the location which was suitable for a it's activity with the user. Furthermore, in Star, in order to refer to the resource

managed by the distributed file system called a file server, the icon called a reference can be created and it can arrange on a 2-dimensional flat surface. By the function of a reference, an alias name can also be given if needed for individual to a shared resource. In addition, although one desktop was prepared to the individual at the beginning as individual work environment, the equipment which prepares two or more 2-dimensional flat surfaces for arranging a resource is also developed.

[0005] What is depended on Macintosh (trademark) of Apple Computer, Inc., and the thing to depend on Rooms (trade name) of Xerox Corp. are known by such conventional technique. In Macintosh (trademark), this technique is applied to the file system by the hierarchical directory. The 2-dimensional flat surface called the folder corresponding to each directory is managed, and the resource which the directory holds can be arranged on it. And if a folder is opened, the display of a 2 more-dimensional flat surface will be obtained. Here, there is a concept of a workspace as a work area characterized by controlling the information which a user is shown according to the change of a task. Bannon and others advocated the concept of the first workspace "it is good to save the environment where it was performed before when the work in office is performing two or more business in order in parallel and it returns to pre- work" in "Evaluation and analysis of users' activity organization" (CHI'83).

[0006] Rooms (trade name) of for example, Xerox Corp. is one of those which realized the concept of this workspace (JP,7-86820,B). Rooms can define the desktop called a room for every business, and the room not only packs the resource, but manages the running state of application. A user can move to the room for other business, saving the condition of the application under activity in each room. Moreover, using two metaphors called a door and an overview, it can connect between sets or a room can also be moved.

[0007] When using two or more environments for two or more operating activities, or in taking over business to other men, it is necessary to recollect or understand the structure and the property of the work which a user performs in the environment. In the above mentioned hierarchical directory and the above mentioned desktop, it can consider as the help which makes a user understand the structure of work by devising the layered structure by the information on resource propers, such as an identifier of a resource, and a class, arrangement in a 2-dimensional flat surface, and the reference to other directories and desktops. For example, the suitable identifier for a resource can be given or the resource related closely can be arranged to near.

[0008] However, it is not so simple as it can be expressed only by the layered structure of arrangement, identifier or directory of a resource, or a folder how a resource is treated in specific business or what kind of relation between



resources is. In Rooms, a room can be made for every business, a resource required for each room can be arranged, and each business can be performed there. However, it being difficult to classify information to compensate for work beforehand, and rearranging a classification according to advance of work requires time and effort very much. Therefore, a setup of a room surely tends to become a rough classification, and there are a system of work and a fault of being not suitable.

[0009] As a conventional technique in which the structure of the information in business can be shown intelligibly for a user, there is invention indicated by JP,60-108975,A, for example. In this invention, a user's directions of the predetermined field of the Shimo pictures image show a user computer resources matched with that field, such as another Shimo pictures image and a document. Two hierarchies of a guide image and an element image express a workspace in this invention. An element image holds one condition of the workspace for working, and a guide image has the semantic information on an element image, and serves as a guide for changing an element image. Therefore, the display of an element image can be changed by choosing one of the tags shown on a guide image.

[0010] Moreover, the two modes are shown in the display of a guide image, and a common image and an individual image can be chosen. By using an individual

image, the framework which treats common information in an individual viewpoint can be prepared. However, it is difficult to build an informational classification system, setting up a guide image beforehand and doing work. Therefore, although it is good when using the classification which was consistent, without separating from the structure of the business set up beforehand, it is difficult to follow in footsteps of the system of the business which changes dynamically.

[0011] Moreover, invention which takes a approach of reusing the use hysteresis of a workspace and building a new workspace is proposed by "U. S.Pat.No.5, 159 and 669, AUTOMATICALLY CREATING A SECOND WORKSPACE OPERATION RECORD INCLUDING HISTORY DATA AND A UNIT ID BASED ON A FIRST WORKSPACE OPERATION", and Trigg et al." In this invention, the card which can hold the link to two or more cards is regarded as a workspace, and it generates as a history workspace which displays the list of a series of actuation hysteresis what kind of card was made on a certain workspace, or what kind of change was made. The part showing the card name under the list becomes support, and the card which appears in the actuation hysteresis list which the history workspace holds can be accessed easily. That is, all the workspaces (card) relevant to a series of actuation will collect as a history workspace, and will be saved. However, although it becomes possible in this invention to save

actuation hysteresis, the left-behind history workspace is the support set for searching the workspace which retrieves the information relevant to the work which cannot call it the place which works about and was performed before and which is sake [ a workspace ] or related.

[0012] Although there are the following in the conventional technique of saving the advance condition of business, these save a scene with the past, it says that programming etc. will be resumed from the scene, and there is no function which holds and reuses the environment of work. There is a function to take discharge of a desktop as a sleep function in Windows 3.1 (trademark). However, it is the purpose to save the present environment and this cannot hold the condition of two or more work. There is a function in which the whole execution environment can be saved as a snap shot dump according to directions of a user in Smalltalk-80. However, the aim of this function is saving the specific condition of work with a snap shot dump, and there is nothing to use that condition at other work. Therefore, two or more work is designed so that it may carry out in another project, and it is essentially the same as Rooms in this viewpoint (Adele Goldberg original paper, Hideo Aiiso supervision of translation, "SMALLTALK-80-interactive programming-environment -", Ohm-Sha, and 1986.). A debugger is the function in which a program execution environment can be saved according to directions or a setup of a user. However, this function

is for saving the advance condition of two or more modules in programming, and the purpose and a configuration differ from this invention which memorizes the maintenance condition of an object.

[0013] Although there are the following in the conventional technique which reuses hysteresis information, these have caught the hysteresis which reuses operation using hysteresis of retrieving information using hysteresis, only with the condition of the mere past. Therefore, the condition in a certain time of the past cannot be restored, and it cannot be said that new work is begun from there. With the coordination activity exchange equipment indicated by JP,6-266662,A, the past work content can be correctly reproduced from the hysteresis of a coordination activity, and it can use for a coordination activity's improvement in efficiency of after that. However, it is only reproducing a work content and the function made into an aim cannot harness the condition of the past work in future work. With the history equipment indicated by JP,6-139117,A, selection of a history is made easy by displaying the screen image at the time of actuation with the history of actuation hysteresis. That is, the bitmapped image of the screen currently then displayed in case the hysteresis of actuation is recorded is incorporated, and in case a history is displayed, a screen image is displayed on coincidence. However, this is only making semantics [ the past ] of actuation easy to remember, and cannot resume work from the condition.

[0014] The circumstances about the node which the user referred to, or its reference path are made to grasp easily in a hypertext document with the document method of presentation indicated by JP,6-214989,A and its equipment. That is, by accumulating node migration hysteresis, displaying the notation with which a node and a link are expressed to the order referred to [ which were referred to and was document-changed ], doubling a hysteresis mark with hysteresis on this, and indicating by migration at the order of time series shows circumstances. However, this only displays hysteresis on time series, and cannot resume work from the condition.

[0015]

[Problem(s) to be Solved by the Invention] In order to share and utilize information systematically, it is important to understand each other information correctly. For that purpose, it is indivisible to get to know the background information from what kind of situation of what kind of work the information held by the in-house was not made. In order to know the background information from which information was made, information needs to be arranged with the view of business. However, it is difficult to classify and arrange information flexibly, an informational classification and an informational system changing dynamically, following in footsteps of this, and advancing business as business advances. From such a situation, the technical problem mentioned to following (1) and (2)

occurred.

[0016] (1) In order for a classification and arrangement of the information which made business suit to manage the information related for every difficulty:business, semantic attachment must be able to be performed in the viewpoint according to each business to information. For that purpose, it doubles with the system of work, and it is necessary to classify and arrange information and to manage it systematically. Moreover, in order to support performing two or more business in parallel, it has two or more environments where the information arranged for every business is offered, and it is necessary to hold an advance condition for every business, and a user chooses the held business freely. Moreover, when the classification of business which was being performed until now tended to be left and it was going to start business from a new viewpoint, the great effort was required for redo of a classification.

[0017] (2) reuse of the know-how acquired through business -- difficulty: -- there is much know-how in work using this document -- that a certain work sends a result to this man, and a certain work carries out such processings -- at a certain work. What can be standardized among these know-how is sharing by manual-izing. However, much know-how acquired through business is difficult to standardize, therefore difficult to share. For this reason, the framework for re-utilizing the know-how acquired through the activation process of such work

was

required.

[0018] This invention was made in view of the above-mentioned conventional situation, and aims at offering the information processor which builds automatically the work environment holding information required for an activity reflecting the activity performed in work environment. Moreover, this invention aims at offering the information processor which arranges information for every business while it builds the above-mentioned work environment. Moreover, this invention aims at offering the information processor which saves the advance situation for every business while it builds the above-mentioned work environment. Moreover, this invention takes out required information for the past condition from the condition of restoration or the past according to work, and aims at offering the information processor which builds the above-mentioned work environment. Moreover, this invention aims at offering the information processor to which initiation of new work is closed using the built above-mentioned work environment if .

[0019]

[Means for Solving the Problem] While considering the above-mentioned work environment as an information-unit set, making into an information unit a data object, an application object, etc. which are used by this work environment and memorizing two or more information-unit sets for an information-unit set

maintenance hysteresis storage means, the information-unit set which becomes an information-unit set maintenance means for an activity is made to hold in the information processor concerning this invention. And when predetermined conditions, such as modification of work, modification of a user, and an addition, deletion of an information unit, are ready, while making an information-unit set maintenance hysteresis storage means memorize the information-unit set which an information-unit set maintenance means holds by the storage directions means, a selection-information grant means gives selection information, such as a work name, a user name, an object name, and time, to the information-unit set memorized to this information-unit set maintenance hysteresis storage means. Moreover, in order to make a certain information-unit set applicable to an activity, when selection information is inputted from a user, the information-unit set which corresponds [ from ] among the information-unit sets the selection means were remembered to be by the information-unit set maintenance hysteresis storage means using the selection information given by the selection-information grant means chooses, and this information-unit set is read from an information-unit set maintenance hysteresis storage means to an information-unit set maintenance means by the read-out means.

[0020] Therefore, in the information processor of this invention, even if it does not classify beforehand according to the activity of bringing together the



information which a user needs at the time in a workspace, according to the advance situation of business, are recording management of the information-unit set (work environment) is dynamically carried out according to it. And since selection information is made to accompany and the information-unit set according to the condition of work of such the past is held, by reading based on selection information, return to the condition at the time of the past and work is started, the past condition can be restored and information required for work can be extracted.

[0021] Moreover, in the information processor concerning this invention, when the number of the information units contained during an information-unit set is changed, if , it has closed trying to save the information-unit set before modification, and restoring an information-unit set in the condition before modification by this always. For example, the sacrifice of the time of the number of the information units contained during an information-unit set starting to decrease from an increment was carried out as a break of work, and if , it has closed restoring the situation in front of the break of this work always. Furthermore, in the information processor concerning this invention, in order to enrich such a function more, it has the following functions.

[0022] (1) It not only holds the condition of the place of current work, but it memorizes transition of the condition of the place of the past work to time series.

(2) The change in the information unit in the place of work etc. supervises various operation in the place of work, and memorize the condition of the place of work to the timing beforehand defined as the Ruhr.

(3) Show a user a state transition using the selection information given at the time of storage of an information-unit set.

(4) If the state transition shown the user to selection information is directed, the condition (information-unit set) of the place of the newest work of fulfilling the condition will be read, and it will change into the condition of the place of current work.

(5) Direct the condition of the place of work and extract the information with which certain conditions are filled from the information unit which the condition holds.

(6) Direct the condition of the place of two or more work, calculate the sum-set of those information units to hold, common class, difference, etc., and create the place of new work.

[0023] In the information processor concerning this invention, in case a user performs two or more business in parallel, the result of natural action of collecting information required for business is made to reflect, and informational classification arrangement is gradually made as business is advanced. Therefore, while those at whom arrangement is not good also do work, it

becomes possible to collect and carry out grouping of the required object one by one, and an informational classification comes to succeed by this. If the comparison with Rooms describes, before working work environment, by Rooms, it is necessary to create but, and also after work progresses, a view is changed and work environment can be recut by this invention. It changes, while an operating system does work, and \*\* memorizes the condition of the place of work to the timing considered to be "a change of work" by this invention with the selection information which needs equipment, doing work. It works in the form where it is as choosing that work name when beginning other work \*\*\*\* [, and ], and with advance of this work, grouping of the information unit which suited the place of the same work will be carried out gradually, and it will be arranged to compensate for work. [ that a user inputs the work name advanced now ]

[0024] Moreover, after having not performed it only when storage of the condition of the place of work had recognized it as the time of the change of work of a user, for example, being automatically carried out to the timing of the change in the information unit in the place of work and having done work, it is possible to branch from transition of the past work and to begin new work. that is, after working, the information unconsciously accumulated by the activity till then is received -- giving the significance (for example, -- or a certain information being used for any work -- a thing with a certain information peculiar to a certain

work -- it being -- information required for work of a certain relation saying [ something ]) -- etc. -- know-how can be extracted and it can reuse to a next activity. Moreover, in this invention, "one certain scene" of the place of work is memorized at any time to the timing which may be the change of work, and attributes, such as a class of work then expressed, a user name, and an inflow resource it is considered that are the trigger of a change of work, are given and memorized. Therefore, if a scene is structured by the class of work, the function same at worst as Rooms can be offered.

[0025] Moreover, in this invention, since each scene of the place of work is saved as "a set of an information unit", an information-unit set can be searched or the new information-unit set by the set operation can be generated. For example, two manuscripts are written by separate work environment, when it is thought that he will make them into one manuscript afterwards, by taking a sum-set, the information which both manuscripts hold is merged and a new manuscript is made. Moreover, when a manuscript which is different by some work environments is being written, a basic environment required in order to write a manuscript can be extracted by taking the common class of those work environments. Moreover, when information has been full of the work environment which is writing the manuscript of a certain theme, supposing there are other work environments which are writing the report about the theme

concerned, information peculiar to the work environment for writing the manuscript of the theme concerned can know something by taking the difference of those work environments.

[0026]

[Embodiment of the Invention] The example of the information processor concerning this invention is explained with reference to a drawing. The configuration of the information processor concerning the 1st example of this invention is shown in drawing 1 . This information processor is constituted by the server 1 and client 11 which were connected through the network 10, and the server 1 and the client 11 are constituted by both with the workstation or the personal computer. In addition, although the implementation gestalt by client-server is taken in this example, it is not necessary to be such a configuration for example, and an information processor may be constituted in one computer, without using a network.

[0027] The server 1 has the information-unit storage section 2, the information-unit set maintenance hysteresis storage section 3, the information-unit set selection section 4, the information-unit set read-out section 5, the selection information grant section 6, and the storage directions section 7, and the storage directions section 7 has the information-unit set storage decision section 8 and the actuation command are recording section 9. In addition, a

server 1 manages the information used in common by two or more users (client).

The information-unit storage section 2 writes in the information unit made in the past, and memorizes it free [ read-out ]. An information unit is for giving semantic attachment in work to a reference object, and has a pointer to a reference object, and additive attributes, such as a comment to a reference object, here.

[0028] The information-unit set maintenance hysteresis storage section 3 writes in two or more information-unit sets memorized in the past with selection information, and holds them free [ read-out ], and the information-unit storage section 2 and the information-unit set maintenance hysteresis storage section 3 are constituted from this example by the relational database. It is the workspace which an information-unit set holds the set of the information unit for referring to a document, application, etc. here, for example, can arrange two or more information units as a list on two-dimensional space (desktop metaphor). The selection information grant section 6 gives selection information, such as a job name, or time, an implementor name, an information-unit name, etc., to the information-unit set memorized by the information-unit set maintenance hysteresis storage section 3.

[0029] The information-unit set selection section 4 selects one of the information-unit sets memorized by the information-unit set maintenance hysteresis storage section 3 based on selection information. The

information-unit set read-out section 5 reads one information-unit set memorized by the information-unit set maintenance hysteresis storage section 3 to the information-unit set attaching part 14 of a client 11. The storage directions section 7 consists of the information-unit set storage decision section 8 and the actuation command are recording section 9, and gives the directions which memorize an information-unit set in the information-unit set maintenance hysteresis storage section 3.

[0030] Judging whether the information-unit set storage decision section 8 memorizes an information-unit set, the actuation command are recording section 9 accumulates the contents of directions of actuation by the user. Namely, the information-unit set storage decision section 8 supervises actuation by the user. Whenever directions called the change of creation and deletion of an information unit, and an information-unit set are performed, an actuation command is accumulated in the actuation command are recording section 9. Furthermore, by referring to the storage activation Ruhr which set up the timing of storage of an information-unit set beforehand, it judges whether an information-unit set is memorized and storage activation is directed to the information-unit set maintenance hysteresis storage section 3. For example, directions of storage activation are performed to the timing which the information unit which an information-unit set holds fluctuated.

[0031] The client 11 has the I/O section 12, the information-unit set display 13, the information-unit set attaching part 14, the information-unit editorial department 15, the selection information list display 16, and the selection information directions section 17. In addition, a client 11 is for a user to work and is realized by the software on a workstation or a personal computer. The I/O section 12 is constituted by I/O devices, such as a display, a keyboard, and a mouse. Here, standard input/output control, such as an icon with the display of the window used common to various kinds of applications, keyboard entry, and cursor and button grabbing, shall be performed.

[0032] The information-unit set display 13 carries out carrying out the display output of the information-unit set held at the information-unit set attaching part 14 to a display screen etc., and is shown to a user. The information-unit set attaching part 14 writes in the information-unit set which is the environment where the current user is working, and holds it free [ read-out ]. The information-unit editorial department 15 changes arrangement of the information unit which the information-unit set attaching part 14 holds according to the directions of a user inputted from the I/O section 12, copy an information unit to other information-unit sets which the information-unit set maintenance hysteresis storage section 3 memorizes, the information-unit set attaching part 14 makes the information-unit set to hold store in the information-unit set maintenance



hysteresis storage section 3, or a comment is added to an information unit.

[0033] The selection information list display 16 carries out the display output of the selection information, such as a job name given to the information-unit set memorized by the information-unit set maintenance hysteresis storage section 3 or time and an implementor name, and an information-unit name, to a display screen. In addition, a user can input directions and can choose the format of selection information shown a list table from the I/O section 12. The selection information directions section 17 directs the conditions for reading from the selection information displayed on the selection information list display 16 as an information-unit set held to the information-unit attaching part 14. In addition, a user can direct the conditions of selection information through the I/O section 12.

[0034] That is, if a user directs the selection information displayed on the selection information list display 16 of a client 11 by the selection information directions section 17, a server 1 will search with the information processor of the above-mentioned configuration the information-unit set which fills all the selection information specified among the information-unit sets memorized by the information-unit set maintenance hysteresis storage section 3. And the read-out section 5 reads to the information-unit set attaching part 14 of a client, and makes it hold about what has the newest date and time of creation in the information-unit set which fills all selection information. And the information-unit

set display 13 carries out the display output of the information-unit set held at the  
information-unit set attaching part 14.

[0035] Here, the concept of a "workspace" required in order to explain the principle of operation of this invention, a "information unit", "an information-unit set", and a "job" is defined. A "workspace" is a concept showing the place of the work for a user. A workspace includes the current information-unit set holding the condition of the place [ of a user ] of current work, and the information-unit set holding the condition of the past of the place of the work. Work in a workspace is carried out by updating a current information-unit set, and can continue and carry out the past work by carrying out the past information-unit set to a current information-unit set. A user may create two or more workspaces for every class of work, and may create one workspace for performing all work.

[0036] A "information unit" is a unit matched with the information in work the significance [ information ] is given with reference to objects, such as information, a document, application, etc. required for work. Information for the object in work of an information unit to give the significance has the arrangement on the comment which the pointer to an object and the user gave, and a workspace etc.

"An information-unit set" expresses the condition at the time of there being a workspace, and is a set of an information unit. The set of the information unit which an information-unit set holds is fluctuated with advance of work. That is,

an information-unit set shows the work environment which the user in a certain time was shown. A workspace memorizes two or more information-unit sets as hysteresis of work, it manages them so that it can restore as a current information unit, and storage of an information-unit set is performed to a trigger in the change of state of the work of the time of the information unit which for example, an information-unit set holds fluctuating. If selection information is directed by the user, one thing which matches a selection condition from the memorized information-unit set will be searched, and it will be considered as a current information-unit set. A user can continue work in the read environment.

[0037] A "job" is a concept showing logical raw [ of the work performed in a workspace ], and a ball. A job is the set of all information-unit sets belonging to the same work. A user's declaration of a job name associates the information-unit set and job which are memorized after it. Conversely, if a job is specified as a selection condition, it will become possible to read the newest information-unit set belonging to the job as a current information-unit set. That is, a job name functions as an index of a change of work.

[0038] The outline of the managed model of transition of the condition of the place of work using the workspace in this example is shown in drawing 2 . The information-unit set attaching part 14 holds the condition of the workspace currently displayed on the current user, i.e., the information-unit set of

KARENTO. For example, it is held on the memory of a client etc. The information-unit set maintenance hysteresis storage section 3 memorizes an information-unit set required at any time in order to enable restoration of the condition of the arbitration of a workspace which changes with advance of work of a user, i.e., the past information-unit set. The storage directions section 7 performs the storage directions to the information-unit set maintenance hysteresis storage section 3 of an information-unit set by trigger called the change in the information unit in the information-unit set which the information-unit set attaching part 14 holds etc. If selection information, such as a job name, an information-unit name, a user name, and time, is directed by the user, the information-unit set selection section 4 will choose the information-unit set memorized by the information-unit set maintenance hysteresis storage section 3, and will read it to the information-unit set attaching part 14 as restoration, i.e., a current information-unit set, by the read-out section 5.

[0039] Thus, the function which gives selection information, and its condition are called to the condition (information-unit set) at the time of there being a workspace, and it restores to it, and has the function to offer the environment which resumes work. By this function, a user can carry out two or more work, changing the condition of a workspace. Moreover, in this invention, a user can carry out the work by bringing together information required for a certain work in

the information-unit set of a specific job name. That is, even if a user does not take an informational classification into consideration beforehand before beginning work, he can arrange an information unit in the viewpoint of work gradually by attaching the job name to the set of an information unit required for work.

[0040] Moreover, a user can declare a job to the timing of arbitration during the activity on a workspace. Declaration of a job is made when a user operates a menu and a carbon button, it is urged to the input of a job name by this, and it is that a job name is inputted by the user and a new job is generated. If a job is declared, the information-unit set maintenance hysteresis storage means 3 will give the job name as selection information of a current information-unit set.

[0041] Moreover, if a user specifies a job by the selection information directions section 17, a system will call the newest information-unit set with a desired job name. And the information-unit set of KARENTO of a workspace is rewritten by the called information-unit set, and a user can resume the work corresponding to a job by displaying it. Thereby, a user can perform two or more work in parallel under the same work environment, changing the information-unit set displayed on a workspace. That is, even if it does not define a room beforehand for every work like Rooms, while performing each work, a job name can be attached to an information-unit set, and the effectiveness of migration to the work environment

corresponding to work can be acquired by changing this.

[0042] The outline of the data at the time of an information-unit set being memorized by the information-unit set maintenance hysteresis storage section 3, or read-out being performed in drawing 3 and the flow of processing is shown.

That is, if a user directs one of the selection information displayed on the selection information list display 16, selection information, such as the job name till then, will be given to the information-unit set of KARENTO, and storage activation of the information-unit set of KARENTO will be carried out to the information-unit set maintenance hysteresis storage section 3. Then, the information-unit set chosen by the directed selection information is taken out on a workspace, The information-unit set turns into a current information-unit set of a workspace, is displayed on an information-unit set display, and can resume work from there.

[0043] The example of the relation between actuation in a workspace and the information-unit set stored by it is shown in drawing 4 in accordance with time series. First, if a workspace is newly created, since manifestation what work to still perform there is not made, immediately after a new information unit flows (that is, work was begun), the input of the class (job name) of the work is demanded from a user. And supposing the job name of "job1" is inputted, the selection information "job1" will be given to the information-unit set memorized

from a degree. Then, when a new information unit flows, the information-unit set at that time is memorized. The information-unit set memorized by the information-unit set maintenance hysteresis storage section 3 is expressed in drawing 4 as "set1."

[0044] Then, the job name of "job1" in all information-unit sets memorized with a subsequent activity is attached until it is performed that the following job name is declared or other jobs are chosen. In this example, by the time a new job is declared, storage of four information-unit sets ("set2", "set3", "set4", "set5") will be performed, and a total of five information-unit sets of job name "job1" are memorized. And when work of "job1" which the user was performing till then is interrupted and then work of "job2" is begun (job name "job2" was expressed), the job name of the information-unit set which memorizes the condition at that time and is memorized further after that is changed.

[0045] In addition, when the restart of work of "job1" is directed again, the newest information-unit set of job name "job1" is searched, the information-unit set turns into a current information-unit set of a workspace, it is displayed on the information-unit set display 13, and, as for a user, the restart of work is attained. Of course, if in charge of storage of a next information-unit set, the job name of "job1" is attached.

[0046] An example of the display on a display screen is shown in drawing 5. In

this example of a display, the information-unit set viewing area 21 which arranges an information unit, and the selection information list viewing area 22 which displays and chooses a job are formed. The information-unit set viewing area 21 arranges information units, such as "Printer", "Doc-4", and "Patent DB", on a two-dimensional flat surface. By directing each information unit with pointing devices, such as a mouse, a user can start the object to refer to, can edit additive attributes, or can change arrangement.

[0047] The selection information list viewing area 22 is a field which displays selection information, such as a job name for choosing an information-unit set, and a user name. There is a thing called the object generated just before performing the time when performing the user when performing storage of the declared job and an information-unit set as selection information and storage of an information-unit set and storage of an information-unit set. In the example of drawing 5, it is set up so that job names, such as "a Database patent" and "an information retrieval patent", may be displayed as selection information. In order that the information-unit set currently displayed on the current information-unit set viewing area 21 may know whether it is a thing about what kind of work, the inverse video of the selection information directed by the user is carried out within the selection information list viewing area 22. By choosing the job name of the selection information list viewing area 22, the information-unit set which the



information-unit set viewing area 21 is shown can be changed. It becomes possible to do two or more work in parallel, directing a job and changing an information-unit set by this.

[0048] Moreover, selection information, such as a user name when memorizing and an information-unit name which flowed, is given to an information-unit set besides a job name. For example, the example which structured and displayed the information-unit set on drawing 6 by the job name and the user name is shown. As for job "Database patent", it turns out that work is done by two persons, user "Aoki" and "Suzuki", and it turns out that job "information retrieval patent" is working by adding "Yamada". Like drawing 6, the newest information-unit set of implementer "Aoki" is called by job name "Database patent" by specifying "Aoki" of "a Database patent" of this list. This is calling the newest situation Mr. Aoki's performing work of "a Database patent", that is, when Mr. Aoki always specifies his identifier, it becomes possible to work without being influenced of Mr. Suzuki's activity. Of course, if Mr. Suzuki specifies "Suzuki", its line is easy to be able to call the newest situation which is and to get to know Mr. Aoki's present advance situation. Moreover, the information-unit set of the newest Database patent can also be called regardless of a user name by specifying job name "Database patent" to know the newest situation of this work regardless of a user.

[0049] Moreover, as shown, for example in drawing 7 , more detailed selection information can be directed now by structuring and showing an information-unit set using the information-unit name which flowed in addition to a job name. For example, if the limit of "limiting and displaying on the information unit which flowed into one week recently" is applied, the advance situation of the latest work can be checked, referring to the situation of the scene of arbitration.

[0050] Next, the concrete DS in this example is shown. Although reference is not made in this example about the version control of the object which an information unit and an information unit refer to, recording all modification of an information unit or its reference place object is also considered by using the version control technique used conventionally. By carrying out like this, the condition of the information unit contained in the information-unit set in the called hysteresis or its reference place object can be restored to the past condition as it is. By realizing such a function, exchange at the time of carrying out writing activities together can be offered.

[0051] An example of the DS for managing a user name (user name) is shown in drawing 8 , and management of a user is attained by the conventional management technique with id, an identifier, and a password. An example of the DS of a workspace is shown in drawing 9 . The workspace has id of the current information-unit set currently displayed on id, the identifier, and the current

workspace field, and id of the display format of a selection information list. A workspace is a subject holding two or more information-unit sets, about how it divides and has a workspace, it uses and there are some directions. For example, usage called one workspace can be done and one person can also do usage of assigning a workspace to a logical task, respectively.

[0052] An example of the DS of a job is shown in drawing 10 , and the job has id, an identifier, the date and time of creation, and an implementor name in it. A job is the identifier of the work which the user declared all over the workspace, and job data are generated whenever a user declares. The DS showing correspondence of a workspace and a job is shown in drawing 11 , and it is the table which expresses correspondence which workspace includes which job, by id. In addition, associated data is added whenever a job is generated. An example of the DS of an information-unit set is shown in drawing 12 . The information-unit set has the information of the information unit id which id(ed) and flowed into the job name of this information-unit set and this information-unit set other than id of works \*-SU which belongs, the date and time of creation, and an implementer.

[0053] An example of the DS of an information unit is shown in drawing 13 , and it has the information of the arrangement on a reference object, starting application, the given comment, and a workspace besides id, an identifier, the

date and time of creation, and an implementer in it. The DS showing correspondence of a workspace and an information-unit set is shown in drawing 14 , and it is the table which expresses correspondence which workspace includes which information-unit set, by id. In addition, associated data is added whenever an information-unit set is generated. The DS showing correspondence of an information-unit set and an information unit is shown in drawing 15 , and it is the table which expresses correspondence which information-unit set contains which information unit, by id.

[0054] Moreover, it has the table of selection information shown in drawing 16 for the display of a selection information list, and the table of the selection information list shown in drawing 17 . It is shown that a job, an implementer, an inflow information unit, and the date and time of creation can use the selection information table of drawing 16 as selection information, and the selection information list table of drawing 17 holds the display format of the selection information list display 16. For example, as the 1st hierarchy's selection information, 1 (from drawing 16 to that is, a job) is set up for selection information id, and, as for the list table of drawing 17 , selection information id shows that 3 (from drawing 16 to that is, an inflow information unit) is set up as the 2nd hierarchy's selection information.

[0055] Next, the Ruhr at the time of memorizing an information-unit set is

explained. The storage to the information-unit set maintenance hysteresis storage means 3 of an information-unit set is performed ignited by actuation series of commands including directions of a user having become a specific combination. A setup of the storage activation Ruhr can perform timing of storage activation with various variations. If a user performs operation on a workspace, the actuation command will be first accumulated in the actuation command are recording section 9. Next, the command accumulated with the storage activation Ruhr is referred to, and a judgment whether storage activation of the information-unit set is carried out is made by the decision section 8.

[0056] A user performs the direct directions "information-unit set storage" with a carbon button etc., or the timing in connection with the change of work of the inflow of the information unit to the all clearance and information-unit set to manifestation of the new job and the information unit on a workspace is detected, and specifically, the actuation command is accumulated in the actuation command are recording section 9, as shown in drawing 18 . The actuation command are recording section 9 accumulates the contents of actuation (CreateUnit) of having deleted the information unit (DeleteUnit), having created the information unit which changed the current information-unit set which opened the workspace for actuation of a user in order (OpenWS), which closed the workspace (CloseWS), and which chose the job (SelectJob) (SwitchJob).

[0057] And the Ruhr for determining whether store an information-unit set is set up as shown in drawing 19 . That is, to the timing of having begun new work or having changed work in the workspace, the condition of the last work is surely saved. When an information unit was incorporated to the information-unit set of the newly created workspace, after deleting all the information units holding a current information-unit set in what is considered as timing of a change of work, when an information unit was generated in it, after specifying the job and changing a current information-unit set, the information unit may have been generated in it. Such timing is considered to be the timing from which new work begins, and storage of an information-unit set is directed to the timing it is considered that is change of such work.

[0058] Next, it explains with reference to the flow chart which shows the processing actuation in this example to drawing 20 thru/or drawing 24 . The procedure of incorporation processing (A) of the information unit which incorporates an information unit to an information-unit set is shown in drawing 20 . Incorporation of an information unit is performed from the file management tool which OS offers by directions by the command or the pointing device. Moreover, incorporation is performed by directions of other applications or there is an approach incorporation is performed by the command of an information processor with a file management tool. If there are directions of incorporation of

an information unit, an information processor will acquire information, such as a user name of the user who performed the directions, its time, and a flowing file name, first (step S1).

[0059] Subsequently, it progresses to the storage decision and executive operation (B) shown in drawing 21 , after generating new data to the data table of an information unit (step S2) and adding "CreateUnit" to the actuation command are recording section 9 (step S3) (step S4). And the newly generated information unit is added to the information-unit set of KARENTO after storage decision and executive operation (B) (step S5). That is, correspondence of id of an information-unit set of KARENTO and id of the newly created information unit is added to the table corresponding to an information-unit set-information unit. Then, regeneration of this information-unit set is performed and processing is ended (step S6).

[0060] The procedure of the storage decision and executive operation of an information-unit set (B) is shown in drawing 21 . This processing will perform storage along the Ruhr, if the accumulated contents of actuation (command) judge whether the storage activation Ruhr is suited and conform. For example, it is called when directions of operation, such as storage directions, read-out, etc. of generation, deletion, migration and edit of the information unit by the user, and an information-unit set, are performed. That is, the condition of the actuation

series of commands accumulated as shown in drawing 18 confirms whether fill the storage activation Ruhr shown in drawing 19 . In this example, sequentially from Rule-1, it checks, and if there are some suiting, storage activation of whether the Ruhr is filled or not will be carried out. Rule-1 shown in drawing 19 expresses "the time of an information unit newly being generated by the newly created workspace." In this case, since conditions will be fulfilled if the command with which the workspace was newly created by the 2nd from the tail of the series of commands accumulated in the actuation command are recording section 9 (CreateWS) that the information unit was made by the tail (CreateUnit) continues, storage activation of an information-unit set is performed.

[0061] That is, when it confirms whether to fill the storage activation Ruhr (step S10) and the Ruhr of storage activation is being filled, it progresses to the new-job registration processing (C) which shows in drawing 22 whether the job name is declared if it checks (step S11) and the job name is not declared about [ having created the workspace ] (step S12). And when the job name is declared, the information-unit set currently held at the information-unit set attaching part 14 is copied to the information-unit set maintenance hysteresis storage section 3, and the selection information of an implementer, time, a job name, and an object name is given (step S13). Subsequently, after changing into a new thing id of the current information-unit set held at the information-unit set attaching part 14



(step S14), id of the newly created information-unit set is added to the table corresponding to a workspace-information-unit set for managing the information-unit set memorized in the workspace (step S15), and processing is ended.

[0062] The procedure of new-job registration processing (C) is shown in drawing 22. From storage decision and executive operation (B), this processing is called, when declaration of the new job is expressed by the user. First, an information processor opens a job name input window to a display screen, and the input of a job name is urged to it to a user (step S20). And if a job name is inputted (step S21), whether the job name concerned is used in the same workspace checks (step S22), and if used, it will return to the step to which it displays on a display screen "The job name is already used", and the input of a job name is urged.

[0063] And when a new job name is inputted, new job data are generated (step S23), "CreateJob" is added to the actuation command are recording section 9 (step S24), and storage decision and executive operation (B) are called (step S25). And the job attribute of an information-unit set of KARENTO which the information-unit set maintenance means 14 holds is changed into id of a new job (step S26), and id of the new job is added to the table corresponding to the workspace-job which manages the job in a workspace (step S27). Then, in order to perform regeneration by the selection information list display 16, it progresses

to classification display processing (D) of selection information shown in drawing 23 (step S28).

[0064] The procedure of classification display processing (D) of selection information is shown in drawing 23. In this processing, the selection information attribute of the 1st hierarchy and the 2nd hierarchy is first acquired with reference to the selection information list table shown in drawing 17 (step S30). And with reference to the table corresponding to a workspace-information-unit set, id of all information-unit sets that a workspace holds is gained (step S31), and those information-unit sets are sorted by the date and time of creation (step S32). Subsequently, a pair of list of the attribute value of the selection information of the 1st hierarchy-2nd hierarchy is generated based on them (step S33), and in order to show the selection information equivalent to the information-unit set of KARENTO, with reference to the current information-unit set of a workspace (step S34), the selection information attribute of the 1st hierarchy-2nd hierarchy corresponding to a current information-unit set is specified (step S35). Then, a pair of list of the attribute value of selection information is displayed (step S36), and the inverse video of the attribute value of the selection information corresponding to a current information-unit set is carried out (step S37).

[0065] The procedure of read-out processing (E) of the information-unit set

performed on the occasion of the change of the condition of the place of work is shown in drawing 24 . This processing will be called if the selection information of the selection information list display 16 is directed by the user, and the directed selection information is gained first (step S40). And "SelectJob" is added to the actuation command are recording section 9 (step S41), and storage decision and executive operation (B) are called (step S42).

[0066] And the newest information-unit set which fulfills the conditions of selection information is searched (step S43), "SwitchSet" is added to the actuation command are recording section 9 (step S44), and storage decision and executive operation (B) are called (step S45). Subsequently, the searched information-unit set is read to the information-unit set attaching part 14 (step S46), and id of the searched information-unit set is made the current information-unit set which a workspace holds (step S47). And in order to perform classification presenting of selection information again, it progresses to classification display processing (D) of selection information (step S48).

[0067] Next, with reference to drawing 25 thru/or drawing 29 , the concrete example of use about this example is explained. The example of a display of the information processor in the start time of this example is shown in drawing 25 . The information processor shows the information-unit set to the large viewing area 21 of the left-hand side in drawing, and the information unit which is the

element serves as an icon for referring to external instruments, such as a document, and application or a printer. Moreover, the field 22 of the right-hand side in drawing is a selection information list display, and selection information is shown by the layered structure of a job name and an inflow object name in this example. As for a current condition, the user has declared four tasks in this workspace, and they are "a Database patent", "an information retrieval patent", "monthly report data", and "report writing." For example, the job of "a Database patent" shows generating five documents, "Doc-1.1", "patent.1", "Doc-3", "test.1", and "Patent-2." Moreover, the condition of the place of the work shown now is, after generating the document "appendix" by the job name of "monthly report data."

[0068] For example, the case where it is going to begin the work which writes a new patent manuscript after this is considered. The patent is a patent about a knowledge database, and suppose it that a technical survey can be used conventionally about "a Database patent" which is a former job. As shown in drawing 26 , the condition of the place of work can be returned to the newest condition of a job name "a Database patent" by choosing a job name "a Database patent" as selection information. In addition, having returned to the condition after generating "Patent-2" finally by the job of a "Database patent" can check by the display of the selection information list 22.

[0069] On the other hand, in the job of a "Database patent", since two patent manuscripts were written, suppose that he wants to use the technical survey used not in the direction of "Patent-2" but in the direction of "patent.1." Then, as shown in drawing 27 , "patent.1" can be chosen from the selection information list 22, and the place of work can be returned to the condition in the middle of the job of a "Database patent." Thereby, the information about "Patent-2" can be eliminated from the place of work. The place of this, i.e., work, is not classified logically and it can attain by having memorized changes of the condition of the place of work. This function enables it to recut branching of the place of work by later different view. Moreover, if a user name is used as selection information, it will become possible to start only the activity done by itself and to also make it branch from there to a new activity from the place of the work which was doing the joint activity.

[0070] Next, in beginning new work from the condition, as shown in drawing 28 , by pushing the carbon button 23 "New Job", the window 24 for declaration of a job name pops up, and it urges declaration of a job name. In this example, the user has inputted "the knowledge DB patent." And a push on "O.K." attaches "a knowledge DB patent" to the information-unit set of KARENTO as a job name. Then, supposing it draws up the document "Presen-1" as shown in drawing 29 for example, "Presen-1" will be added to the selection information list display 22

as "a knowledge DB patent" and an object name as a job name.

[0071] The difference between the description of this invention and the conventional technique is explained. From Rooms, the room of a "Database patent" is made first, and it works by it there. And when it is thought that he will perform the next work, the room "an information retrieval patent" will be made and a new environment will be built there. Here, in order to use a printer required for writing of a patent manuscript, an editor and application called Patent DB, or the document referred to in common for both of the work, it will be copied by Rooms. However, since this activity needs to copy a required information unit altogether between rooms, it will become complicated. Therefore, writing of two or more similar business, for example, two or more patent manuscripts like this example, etc. will tend to use the same room.

[0072] On the other hand, in this invention, since the form where name the condition in a certain time and it is called is taken unlike newly making a room, the time and effort which builds a similar room can be saved, and work can be subdivided easily. That is, if work is made to derive, the front condition remains as it is. A workspace is in the same condition as the final state of a front job immediately after declaring a new job. Then, the unnecessary thing is tidied up to the following job (getting it blocked and what is not needed eliminating), required information units are collected, and a new job environment will be

prepared. Rooms seems to just have many desks for every work, when this is compared to the desk in the real world. On the other hand, this invention takes the snap shot dump of the condition of the desk which is working at any time, attaches the identifier of work, and returns it to the condition of having done the work for the condition of a desk before, by declaring the identifier.

[0073] There are some which are naturally used also in common with which work in a desk, and there is also a changing thing the significance [ thing ] is given by work. It is difficult to get to know the description of each of these information units beforehand, and it is realized in this invention that nature and an information unit are classified, doing work. Moreover, by this invention, when you think that he will work "a Database patent" again after having tidied up the unnecessary thing by the job of an "information retrieval patent", a front condition is restored only by specifying the job name of a selection information list. In addition, this invention is not a thing aiming only at restoring the past for the purpose of making a new condition using the past condition. Of course, if it is made to perform version control of the object itself, it is also possible to return the edit condition of a document at the storage stage of works \*-SU, and it is effective when supporting co-authoring etc.

[0074] Various following modification can be added to the 1st above-mentioned example, for example. The storage directions section 7 may not have the

actuation command are recording section 9, but it may constitute it so that the information-unit set storage decision section 8 may perform storage activation according to storage directions of a user. Moreover, when it does not have the actuation command are recording section 9, but the information-unit set storage decision section 8 supervises actuation by the user and directions of creation and deletion of an information-unit set are performed, the storage directions section 7 may be constituted so that storage activation may be performed. Moreover, the information-unit set display 13 may be displayed two-dimensional or in three dimension using a substrate image, and even if it sorts in an alphabetical order or generation time of an identifier of an information unit etc. and is shown a list table, it is not cared about.

[0075] moreover, as the implementation approach about hysteresis storage of an information-unit set To the timing which the information-unit set which the information-unit set attaching part 14 holds, and the newest information-unit set memorized by the information-unit set maintenance hysteresis storage section 3 are made always in agreement, and memorizes an information-unit set The information-unit set in the hysteresis storage section 3 may be copied, and the approach of giving new id to the information-unit set which copied, and reading it to the information-unit set attaching part 14 may be used. Moreover, although [ the above-mentioned example ] the condition of the last work is surely saved as



the implementation approach about hysteresis storage of an information-unit set to the timing which changed work, the approach of saving the condition of work immediately after surely changing work may be used.

[0076] Moreover, it is also considered that the information-unit storage section 14 manages the version of an information unit. The existing version control technique can attain this easily, and the information-unit storage section 14 manages the set of each version of an information unit with updating time in this case. Thereby, when an information-unit set is restored, the information which an information unit holds can also be returned to the condition at the time of memorizing the information-unit set. Moreover, the object management section is prepared and managing the version of the object of the reference place of an information unit is also considered. The existing version control technique can attain this easily, and the object management section manages the set of each version of an object with updating time in this case. Thereby, when an information-unit set is restored, it can return to the condition at the time of the object of the reference place of an information unit also memorizing the information-unit set.

[0077] Moreover, the storage activation Ruhr directions section is prepared and it is possible to customize the storage activation Ruhr of an information-unit set in accordance with the aspect of affairs of use. As timing of storage activation,

the timing it can be considered that is change of work is suitable, for example, the location of the timing which changes the attribute of an information unit, for example, an information unit, is changed, or supposing the information which expresses the advance situation (for example, under un-starting / advance statuses, such as - completion) of work to an information unit is attached, the timing which changes it can be used. Moreover, the storage directions section 7 does not have the actuation command are recording section 9, but it is also considered that the information-unit set storage decision section 8 memorizes an information-unit set periodically for every fixed time amount. Thereby, it can leave the outline of the flow of work. Moreover, if the information-unit set storage decision section 8 carries out the fixed number inflow of the information unit, taking the approach of memorizing an information-unit set will also be considered. Thereby, it can also leave hysteresis for every advance of a fixed activity. Furthermore, preservation of the situation thinned out moderately can be performed and it can leave the outline of the flow of work, without being influenced by the roughness and fineness of work.

[0078] Moreover, the check of the storage activation Ruhr about storage decision of an information-unit set may be performed sequentially, and when checking Rule-1, you may confirm in parallel whether other Ruhr fills. Moreover, first, with reference to the actuation command are recording section 9, you may

lengthen reverse from actuation of a tail, and may also narrow down the Ruhr which may be filled. Moreover, the actuation command are recording section 9 may accumulate an event which is listed not only to actuation by the user but to a degree, and may judge whether the information-unit set storage decision section 8 memorizes based on it. For example, the event of other applications on the same computer that other applications started or other applications accessed external devices, such as a network and a printer. An event called arrival or dispatch of an electronic mail. Events, such as arrival and dispatch of the work in the workflow tool which is working on the same computer. in addition -- all -- others -- the timing of a change of work can be caught more in a detail by incorporating the event which application generates.

[0079] The configuration of the information processor concerning the 2nd example of this invention is shown in drawing 30 . In addition, the explanation which gives the same sign to the 1st example and intersection which were mentioned above, and overlaps is omitted. By this example, it compounds by carrying out the set operation of the information unit which two or more specified information-unit sets hold, and the example which generates a new information-unit set is explained. In addition to the configuration of the 1st example, the server 1 has the information-unit set composition section 31 which performs the set operation of the information unit which two or more

information-unit sets hold, and the information-unit extract section 32 which filters the information unit which an information-unit set holds. The information-unit set composition section 31 performs set operations, such as a "sum-set", "common class", and "difference", to the information unit which two or more selected information-unit sets hold. The information-unit extract section 32 applies various filtering to the information unit which the selected information-unit set holds with attributes, such as implementer, date-and-time-of-creation and comment / starting application, arrangement, etc. which the information unit holds, and generates an information-unit set new as the result.

[0080] In addition to the configuration of the 1st example, the client 11 has the information-unit set composition directions section 35 which gives directions of a set operation, and the information-unit set extract directions section 36 which gives directions of filtering. The information-unit set composition directions section 35 is a module for specifying two or more information-unit sets for compounding, and directing a synthetic approach to it, as shown in drawing 32 . For example, the window 41 shown in drawing 32 starts by pushing the "synthetic" carbon button 25 shown in drawing 31 . And the information-unit set composition directions section 35 has the function direct "sum-set" 42, "common-class" 43, "difference" 44, and "filtering" 45 to the selected

information-unit set, and a user can specify two or more information-unit sets, and it can give the directions with which the set operation of a "sum-set" or "common class" is performed to the information unit which the information-unit set as which it was specified holds. Moreover, a user can specify one information-unit set used as criteria, and others' one or more information-unit sets, and can give the directions which perform the set operation of "difference."

[0081] The information-unit set extract directions section 36 calls the window 51 shown in drawing 33 by pushing "filtering" carbon buttons 26 and 45 shown in drawing 31 or drawing 32 . By inputting conditions into this window 51, a user can extract the information unit which an information-unit set holds. For example, if its user name is put in and filtered to an "implementer", all the information units that users other than themselves created will be deleted from the information-unit set of KARENTO. Moreover, when called from the synthetic directions window 41 shown in drawing 32 , the information unit to compound will be beforehand extracted to filter conditions, and a synthetic operation will be performed. For example, if starting application is used as the editor which he is using as filter conditions, it can extract to the "document" which can be referred to by the editor, and an information-unit set can be compounded.

[0082] In addition, filtering of an information unit can be directed on the conditions which specify the range of the creation date of an information unit as

"time" and which are extracted to an information unit including the comment specified as a "comment" of extracting to the information unit arranged in a certain range on two-dimensional space as "arrangement." Moreover, filtering by the attribute whether the class of resource which can filter also except the example shown in drawing 33 , for example, an information unit refers to is a "document", and whether it is "application" or the attribute of a resource to refer to, the character string which the resource to refer to contains is also considered.

[0083] The concept of the synthetic operation of a sum-set, common class, and difference is shown in drawing 34 . A "sum-set" is used for integration of two or more work, and "common class" can be used for the extract of an environment common to two or more work, and it can use "difference" for the extract of a resource peculiar to a certain work. In a sum-set, if it  $A = \{a, b, c, d\}$  and  $B = \{a, c, d, e, f\}$  set = Comes to carry out the OR operation of two or more information-unit sets, it will become  $A+B = \{a, b, c, d, e, f\}$ . Moreover, in common class, an AND operation is performed and it is  $A*B = \{a, c, d\}$ . Moreover, the difference of A and B is what deducted the information unit contained in B from the information unit contained in A, and is  $A-B = \{b\}$ . In addition, what is not contained in A and contained in B is disregarded by this operation.

[0084] Next, with reference to the flow chart shown in drawing 35 thru/or drawing 38 , the processing in connection with the sum-set of an information-unit set,

common class, difference, and filtering is explained. The procedure of processing in the case of taking a sum-set is shown in drawing 35 . First, if an information-unit set of some [ the synthetic directions screen 41 shown in drawing 32 ] is chosen and the sum-set carbon button 42 is pushed, id of the selected information-unit set will be acquired (step S50). For example, "a Database patent", "an information retrieval patent", and a "information share conclusion" are chosen for a job name, and the newest information-unit set is searched with drawing 32 , respectively. Subsequently, id of the information unit which each searched information-unit set holds is acquired from the table corresponding to an information-unit set-information unit (step S51), and the set of an information unit id is generated about each information-unit set.

[0085] And id of those information units is merged, processing in which the same thing is deleted is performed, the sum-set of a set of each information unit is taken, each information unit contained in a sum-set is copied, and the new information unit id is attached to each (step S52). Subsequently, a new information-unit set is generated (step S53), and all the information units that copied to the table corresponding to an information-unit set-information unit at step S52, and attached id are added (step S54). Then, id of a new information-unit set is added to the table corresponding to a workspace-information-unit set (step S55), and regeneration of an

information-unit set is performed (step S56).

[0086] The procedure of processing in the case of taking common class is shown in drawing 36 . First, if some information-unit sets are chosen and the common-class carbon button 43 is pushed on the synthetic directions screen 41 shown in drawing 32 , id of the selected information-unit set will be acquired like the example of a sum-set (step S60), id of the information unit which each information-unit set holds will be acquired from the table corresponding to an information-unit set-information unit (step S61), and each set will be made. Subsequently, the AND set of all information units is taken, each information unit of the AND set is copied, and the new information unit id is attached to each (step S62).

[0087] And a new information-unit set is generated (step S63), and all the information units that copied to the table corresponding to an information-unit set-information unit at step S62, and attached id to it are added (step S64). Then, id of a new information-unit set is added to the table corresponding to a workspace-information-unit set (step S65), and regeneration of an information-unit set is performed (step S66).

[0088] The procedure of processing in the case of taking difference is shown in drawing 37 . first, some information-unit sets choose on the synthetic directions screen 41 shown in drawing 32 -- having -- difference -- if a carbon button 44 is



pushed, id of the selected information-unit set will be acquired like the example of a sum-set (step S70), and id of the information unit which each information-unit set holds will be acquired from the table corresponding to an information-unit set-information unit (step S71). Subsequently, the input of an information-unit set used as the criteria of a synthetic operation is demanded from a user (step S72). For example, the display output of all the information-unit sets checked with the message "which is the information-unit set used as criteria" is carried out to a display screen. In the example of drawing 32, three information-unit sets, "a Database patent", "an information retrieval patent", and a "information share conclusion", will be chosen, a user will choose one of them, and the difference on the basis of it will be taken.

[0089] For example, when it chooses on the basis of "a Database patent", processing except what is contained in the list of information units id which other information-unit sets (that is, "an information retrieval patent" and the information-unit set of the newest respectively of a "information share conclusion") hold from the list of information units id which the information-unit set (that is, the newest information-unit set of a "Database patent") used as (step S73) and criteria holds is performed. And all the information units that remained are copied and the new information unit id is attached to each (step S74). Next, a new information-unit set is generated (step S75), and all the information units

that copied to the table corresponding to an information-unit set-information unit at step S74, and attached id to it are added (step S76). Then, id of a new information-unit set is added to the table corresponding to a workspace-information-unit set (step S77), and regeneration of an information-unit set is performed (step S78).

[0090] Next, the procedure of processing in the case of filtering is explained. The filter condition input screen 51 shown in drawing 33 is displayed by pushing the filtering carbon buttons 26 and 45 in the screen of drawing 31 or drawing 32 . The function of filtering is divided into the following two by the difference of this call direction.

(1) The function to restrict the information unit for composition on filter conditions when compounding as mentioned above by the case where it is called from the synthetic directions screen shown in the function which filters to a current information-unit set by the case where it is called from the information-unit set display screen shown in drawing 31 , and (2) drawing 32 .

[0091] The procedure of the filtering processing about (1) of the former is shown in drawing 38 . If the O.K. carbon button is pushed by the filter condition input screen 51 shown in drawing 33 , the directed filter conditions will be acquired (step S80). Subsequently, id of the information unit which acquires id of a current information-unit set (step S81), and a current information-unit set holds from a

workspace table is acquired from the table corresponding to an information-unit set-information unit (step S82). And filter conditions are applied to each information unit (step S83). For example, if filter conditions are "implementer =Aoki(s)", Implementer id will choose only the information unit which is a user name Aoki.

[0092] And all the information units that copied all the selected information units, attached the new information unit id to each (step S84), generated a new information-unit set (step S85), copied to the table corresponding to an information-unit set-information unit at step S84, and attached id are added (step S86). Then, id of a new information-unit set is added to the table corresponding to a workspace-information-unit set (step S87), and regeneration of an information-unit set is performed (step S88).

[0093] On the other hand, when performing the synthetic operation of a sum-set and common class, and difference about (2) of the latter, an operation is attained by performing the same filtering as (1) to the set compounded and obtained.

[0094] Next, a use is explained using the concrete example of the operation of "a sum-set (superposition)", "common class (multiply)", and "difference", and the combination of "filtering." "A sum-set (superposition)" is an operation which takes a sum-set to the information unit which two or more information-unit sets hold. It is effective when unifying two or more work, and this is used. For

example, two patent manuscripts are written by another work environment, when it is thought that he will make them into one patent manuscript afterwards, by taking a sum-set, the information which both conditions hold is merged and a new environment is made. If an example describes, two patent manuscripts are written by another job, and when it is thought that he will write by making them into one patent manuscript, the information-unit set of the newest respectively of the job of one patent manuscript and the job of the patent manuscript of another side will be piled up. Thereby, the information unit of a proper is merged and displayed on each job as an information unit common to both jobs.

[0095] During an information-unit set 1={printer, an editor, the reference patent A, the reference patent B, the reference patent C, and writing, during a document 1 and writing For example, document 2}, Although a patent manuscript which there are two information-unit sets document 5} during an information-unit set 2={printer, an editor, a graphic tool, the reference patent A, the reference patent D, the reference patent E, and writing during a document 3 and writing during a document 4 and writing, and is different by each was written It can pile up, when it plans to make it one patent at a certain time. the information-unit set made to repeat mutually -- information-unit set  $(1+2) = \{ \text{-- under a printer, an editor, a graphic tool, the reference patent A, the reference patent B, the reference patent C, the reference patent D, the reference patent E, and writing -- it becomes with$

document 5} during a document 4 and writing during a document 3 and writing during a document 2 and writing during a document 1 and writing.

[0096] By using this function, which information unit is used in common and composition of an information-unit set is attained, without being conscious of which information unit saying whether to be the thing of a proper. If "application" is chosen as conditions for filtering at this time, it becomes information-unit set  $(1+2) * ("application") = \{\text{a printer, an editor, and a graphic tool}\}$ , and all the applications used with a patent manuscript can be extracted. In addition, although this example showed the example which piles up two information-unit sets for example, when having scoured the idea in two or more tooth spaces, respectively (with two or more persons), compounding it and making a proposal Comment attachment to collection and it of reference is performed in two or more tooth spaces, respectively (with two or more persons), composition of many information-unit sets is more needed, and the effectiveness of this function is more conspicuous for the time of summarizing them.

[0097] "Common class (multiply)" is the function to extract a secondary information unit common to two or more snaps. For example, suppose that the patent manuscript was written by some jobs. By multiplying the snap of those jobs, a secondary information unit required in order to write a patent manuscript is extracted. Thus, the fundamental environment for writing a patent manuscript

can be acquired. If it multiplies in the example of the above-mentioned information-unit sets 1 and 2, the new information-unit set information-unit set  $(1*2) = \{\text{a printer, an editor, and the reference patent A}\}$  will be extracted. This can be made into the template of the information-unit set for writing a new patent manuscript. For example, the case where I am allowed to refer to the workspace of those who are writing many patent manuscripts is considered. Since it is the workspace which others are using, it is unclear in what suited that he wanted to do which environment. Some job names which are likely to relate to for example, a patent manuscript can be chosen there, and the minimum environment currently used by performing those crossing when the man always writes a patent manuscript, or the document always referred to can extract the know-how of something. Furthermore, the information-unit set in which the know-how was made to reflect can be generated, and work can be begun there.

[0098] "Difference" is a function for extracting information peculiar to a snap to crossing and reverse. If the snap of a primitive state is lengthened from the snap which is performing a certain work, the information on a proper can be started to the work as a result. For example, suppose that the information-unit set information-unit set 3=  $\{\text{a printer, mail, an editor, a database, a memo pad, a clock, an idea note, and a schedule book}\}$  was used as a normal state. And when the information unit of a proper is extracted in the patent manuscript under

writing by the information-unit set 1, it turns out that it is the information unit used in order that document 2} may be chosen during a document 1 and writing during information-unit set (1-3) = {reference patent A, the reference patent B, the reference patent C, and writing and these may write this patent manuscript. If this function is used, when it will not be clear anymore which information unit is the need when similar work is being performed by two or more jobs, it can arrange by extracting the essence of that job by taking difference. moreover -- for example, there is work environment which is writing the patent manuscript of a certain invention, and supposing the work environment which is writing the report of the same theme temporarily suits elsewhere, information peculiar to the work environment for writing a patent manuscript can know something by taking the difference of those work environments.

[0099] In addition, although information units are matched in case these operations are performed, in this matching, the decision approach whether that information unit is the same can consider matching these both sides other than the identity of id of an information unit shown in the example, concerning the identity of an information-unit name, and the identity of a reference object. The criteria of this identity give a user some alternative, and it is also considered that the user itself chooses the matching approach.

[0100] Here, the difference with the conventional technique and this invention is

explained. The user needed to classify information according to systems, such as an old folder and Rooms, to compensate for work beforehand. On the other hand, in this invention, information can be arranged for every work, doing work. Therefore, systematization of the flexible information doubled with work is attained. That this becomes possible distinguishes and memorizes "the condition of the place of current (one) work", and "two or more conditions of a place of the past work" by this invention to holding "the current condition of the place of two or more work" at Rooms. Thereby, two next features are acquired.

[0101] (1) A degree of freedom is to relate the condition of the place of current work with what kind of work, and memorize it. Therefore, after doing work to some extent, it becomes possible for the work performed until now to attach and save the selection information related with work different from the former.

(2) By giving two or more attributes to the condition of the place of the past work, the alternative of the condition of the past in which read-out in the condition of the place of the present work is possible can be changed. Although cannot distinguish a condition and it can be saved only in the name of a "room" in Rooms, for example, in this invention If two attributes, such as "an identifier of work" and a "user name", are attached to the condition of the place of work It also becomes possible to call the condition that could use the condition of the place of the newest work of the identifier of the work as the present condition



when specifying "in the name of work", and specified both "the identifier of work", and the "user name", for example, a one (or other someone) was performing the work before.

[0102] Moreover, the greatest difference from Trigg's and others history workspace patent leaves the hysteresis of work, and is that the place of the work is reusable. Actuation hysteresis constitutes a workspace new as a list from Trigg's and others history workspace. Thereby, use of being as finding information from the hysteresis \*\*\*\* [ , and ] is possible for a user. [ referring to the process of the work performed before ] That is, a history workspace shows a user the relation between workspaces by making actuation hysteresis of work a list. However, although the call relation between workspaces is saved in a history workspace, in this invention, the contents to save differ and the condition of the object actuation performed on the workspace and a workspace is saved. That is, this invention saves information more detailed than a history workspace, and enables thereby more detailed scene reappearance. Furthermore, it becomes possible for this to begin new work using the condition of the called past.

[0103] Moreover, invention about preservation of an advance condition or reuse of hysteresis differs from the thing of the former [ viewpoint / of hysteresis use ]. The system or know-how of work which are only reuse of operation and can be

extracted from there as generation of the new work environment using hysteresis information are not utilized enough. On the other hand, an object is classified according to this invention into the group of the object which a scene holds by considering that each past scene is each set rather than considering that hysteresis is one scene of the mere past. Moreover, by being able to perform reference and reuse of related information or know-how, and saving two or more conditions of a workspace of performing the same work, this invention can refer to the condition and related information of the past work, or can return to the condition of the past work and can begin new work. That is, since the hysteresis performed on the workspace is accumulated, some [ or ] information which a certain work is sufficient as some information required for a certain work, and is used can extract and reuse the know-how of work of something in the information on a proper at a certain work.

[0104] The configuration of the information processor concerning the 3rd example of this invention is shown in drawing 39 . In addition, this example corresponds to invention mainly indicated to claim 24 - claim 37. The are recording section 61 which memorizes two or more places (Place) which did grouping of two or more information-unit sets to the information processor of this example further, The current condition attaching part 62 which holds two or more information-unit sets used as the candidate for a display with the selection

information, The current status-display section 63 which carries out the display output of the thing corresponding to the selection information inputted by the user of the information-unit sets currently held at the current condition attaching part 62 to a display screen, The related condition extract section 64 which extracts the information-unit set corresponding to the extract regulation about the selection information set up beforehand from the are recording section 61, The related condition attaching part 65 holding the extracted information-unit set concerned, and the list display 66 concerned which carries out the display output of the list of the selection information of the extracted information-unit set to a display screen, While judging whether the information-unit set currently held at the related condition acquisition section 67 which reads the specified information-unit set from the related condition attaching part 65, and is made to hold to the current condition attaching part 62, and the current condition attaching part 62 is saved in the are recording section 61 according to the regulation set up beforehand It has the current condition induction 69 and \*\* which make what was chosen by the user of the Place(s) currently held at the preservation decision section 68 which performs preservation processing, and the are recording section 61 hold to the current condition attaching part 62.

[0105] Here, in this example, the information unit mentioned above is made into the object called a mediator (Mediator), and the information-unit set is made into

the object called a snap (Snap), and Snap of further some is summarized as an object called a place (Place). Moreover, the are recording section 61 of this example is mostly equivalent to the information-unit set maintenance hysteresis storage section 3 mentioned above, and the related condition induction 67 is mostly equivalent to the information-unit set read-out section 5 mentioned above. Moreover, the current condition attaching part 62 of this example is mostly equivalent to the information-unit set attaching part 14 mentioned above, and consists of this examples further as what is equipped with the current status-display section 63 and the current condition acquisition section 69 as a function of the information-unit set attaching part 14. Moreover, the preservation decision section 68 of this example is mostly equivalent to the storage directions section 8 mentioned above, and is further considered as the configuration which also has the function of an are recording means by which the preservation decision section 68 makes an information-unit set (Snap) memorize to the are recording section 61 by this example. Moreover, the related condition extract section 64, the related condition attaching part 65, the list display 66, and a related information assignment means to mention later are mostly equivalent to the information-unit set selection section 4 mentioned above, and the information-unit set (Snap) corresponding to the selection information directed by the user is chosen.

[0106] In addition, as for the information processor of this example, only the are recording section 61 is put on a server side, other modules 62-70 consist of server client formats put on the client side, and all of a server and a client consist of a workstation or a personal computer. However, in this invention, it is very good in the method which exchanges the information which it is not necessary to take the gestalt divided into the server client for example, the are recording section 61 is also put on a client side, and the are recording section 61 holds among two or more clients.

[0107] The above-mentioned are recording section 61 is a relational database, and has a data base schema called Place, Snap, Job, Mediator, User, and IDMax as shown in drawing 40 (a) - (f). Place is an object representing the location which does some work, and has the set (listSnap) of the newest condition of two or more work environments (Snap) currently performed in id (user) of the identifier (placeID) of Place, generation time (createDate), and the user that generated, the identifier (placeName) of Place (location), and its location in the group. for example, Place = {-- one -- ' -- Wed Jul -- 1015 -- : -- 00 -- : -- 00 -- JST -- 1996 -- ' -- 1111 -- ' -- Nomura home -- ' -- ' -- {-- one -- two -- six --} -- ' --} -- \*\* -- having said -- a configuration -- becoming -- \*\*\*\* .

[0108] Snap is an object which shows the condition of work environment, and has the set (mediators) of the information unit (mediator) which id (user) of the

identifier (snapID) of Snap, generation time (createDate), and the user that generated, id (myJob) of Job which shows the work currently performed by the Snap, id (myPlace) of Place from which the Snap was made, and its Snap hold in the group. For example, Snap = It has composition, such as {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, and '{1, 3, 4}}'.

[0109] Job is an object showing the class of work and functions as a label for choosing Snap. Job has id (user) of the identifier (jobID) of Job (work), generation time (createDate), and the user that generated, and the identifier (jobName) of the work in the group, for example, is Job =. It has composition, such as {1, 'Wed Jul10 15:00:00 JST 1996', 1111, and 'CSCW survey'}. In addition, even if the identifier of work is the same, if it is another object, it will be distinguished as another work. Therefore, in order to work together by two or more Place(s), Job needed to be copied and this copy is realized by introducing Snap into Place in this example.

[0110] Mediator is an object showing an information unit. The identifier of Mediator (mediatorID), Generation time (createDate), a user's generated id (user), The reference point addresses, such as URL (Uniform Resource Locators) which the identifier (myName) of Mediator and Mediator point out (myLocation), It has in the group the memorandum (myMemo) which Mediator holds, the X coordinate (myMX) as which Mediator is displayed, and the Y

coordinate (myMY) as which Mediator is displayed. For example, Mediator = It has composition, such as {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 'report draft', 'http://abc.www.abcdef.co.jp/', 'This is the first draft of the report.' (nomura), 1, and 3}.

[0111] User is an object showing a user and has the identifier (userID) of User, generation time (createDate), the generated user name (userName), a user's password (password), and the identifier (homePlace) of Home Place in the group. For example, User = It has composition, such as {1111, 'Wed Jul 10 15:00:00 JST 1996', 'nomura', '\*\*\*\*\*', and 1}. Here, Home Place is Place which the user mainly uses, for example, is Place used as the address for delivery of Mediator when it is opened at the time of a log in or Mediator is delivered by the user with an electronic mail etc. from the outside of a system.

[0112] IDMax is a table which enables it to take within a database the coordination of id (identifier: use it in the semantics of the label in which YUNIKUNESU of an object is shown), also when two or more clients generate Place, Snap, Job, or Mediator to coincidence, for example, it is IDMax =. It has composition, such as {3, 12, 7, 23}.

[0113] When maintenance of this coordination was explained and a client generates one Mediator, id of that Mediator performs first processing set as the temporary thing. First of all, in case the role of id reads information from a

database and generates an object, it is for restoring the relation between objects, and within the same session of a client, when id is unnecessary since reference of Mediator etc. is performed per object, and id is stored in a database, it only needs to be assigned. then, the number of Mediator(s) which the client searched all Mediator(s) in opened Place, and got to know the number of newly generated Mediator(s), acquired the value (this example 23) of "mediator" of IDMax, and were newly generated to this when a preservation instruction in a database was performed from a user -- a guide peg -- it updates with a number the bottom. Thus, after reserving id, even if it generates an object to coincidence by two or more clients by actually assigning and saving id, the batting of id does not happen. In addition, the decision approach of id is very good in the approach of it not being necessary to restrict to such an approach, and calculating dynamically combining parameters, such as generation time and User id.

[0114] The data which consist of Place, Snap, above-mentioned Job, and above-mentioned Mediator are held by the model building as shown in drawing 41 (a) at the present condition attaching part 62, and are held by the model building as shown in drawing 41 (b) at the are recording section 61. That is, data are held per Place at the current condition attaching part 62, and a user is shown the data of this Place unit. Here, Place holds the condition of two or more work done in the current spot place, and is directing two or more Snap(s) with



maintenance or a pointer as DS. in addition, every -- Snap has pointed out Job, respectively and Place has also become pointing out two or more Job(s) as a result. On the other hand, within the are recording section 61 (database), Place, Snap, Job, and Mediator are managed separately, respectively, share each other id(s) and are connected.

[0115] In this example, the above-mentioned present status-display section 63 has the display screen as shown in drawing 42 , and serves as an interface which receives the input from a user with a display output. Each part of the screen concerned has the following functions. If a log in is performed by pushing the log in carbon button 71 after inputting a login name into the log in field 72 and a log in is performed, the user's home Place will be displayed as a default of the Place selection menu 74. Moreover, if the opening carbon button 73 is pushed, Place chosen by the Place selection menu 74 will be opened, Job which the Place points out will be displayed on a job list, and the set of Mediator which Snap which points out the job of the head of a job list holds will be displayed on a snap view. The list of all Place(s) that can be chosen is displayed on the Place selection menu 74, and a user can open Place of arbitration on a screen by choosing one of these.

[0116] A push on the Save carbon button 75 gives the directions which write out and save all the information updated after opening Place in a database 61.

Moreover, if the CallSnap carbon button 76 is pushed, a screen display of the window shown in drawing 44 will be called and carried out, and the directions for extracting Snap stored in the database 61 in this window will be given by the user. When the ViewPile carbon button 77 is pushed, a screen display of the related condition window shown in drawing 43 is called and carried out, and the attribute of the Snap set which the related condition attaching part 65 holds by this window is shown a list table.

[0117] A push on the Combine carbon button 78 calls and carries out a screen display of the window as shown in drawing 59 mentioned later. In addition, if the directions which make two or more Snap(s) compound in this window are inputted by the user, based on this, predetermined set operation processing will be performed by the condition composition section so that the 4th example mentioned later may explain. If the New carbon button 79 is pushed, processing which makes Mediator newly generate is performed, in this new generation processing, the character string written to the Mediator name field and the URL field will be read, and processing which makes it the attribute of new Mediator will be performed. If the Cut carbon button 80 is pushed, it will delete from Snap which makes Mediator the current processing object, and further, if the Copy carbon button 81 is pushed, processing which makes the Mediator concerned hold to a copy buffer will be performed.

[0118] When the Paste carbon button 82 is pushed, processing which pastes Mediator which Cut(ed) or Cop(ied) is performed, and Mediator Cut(ed) or Cop(ied) is Paste(d) by the same coordinate as X and the Y coordinate which existed origin. In addition, when other Mediator(s) already exist in the coordinate Processing in which Mediator is arranged on that right-hand, and it arranges on right-hand further when other Mediator(s) exist also in this location is performed repeatedly. As this result When it has reached to the right end of the display screen, while advancing one Y coordinate, X coordinate is returned to a left end, the same activity is repeated, and processing arranged in a location with an opening is performed. The snap view 83 is a field which displays Mediator by the icon 84, and Mediator which Snap which points out Job chosen by the job list 89 holds is displayed on this field. As for these-displayed Mediator, selection or memorandum, and reference of a reference are performed by pointing devices, such as a mouse. Moreover, migration of Mediator is performed by the drag and drop of a pointing device.

[0119] The Mediator icon 84 is an icon showing Mediator, and the display of an icon is changed according to the class of reference. For example, in Windows (trademark of Microsoft), the extension of ".txt" etc. regular for every format is assigned, and a different icon for every format can be displayed. Moreover, Perth of the character string of URL is carried out, and if it is URL in the company

and is an in-house document icon and external URL, distinction called an external document icon is also possible. The Mediator name field 85 is a field where the identifier of Mediator chosen in the snap view 83 is displayed, and the URL field 86 is a field where URL of Mediator chosen in the snap view 83 is displayed.

[0120] when the NewJob carbon button 87 is pushed, the identifier (character string) inputted into the NewJob name field 88 by the user is new -- processing which generates Job is performed. in addition -- this -- new -- when the NewJob carbon button 87 is pushed, Snap which points out Job copies Snap by which it was indicated by selection, and is generated. For this reason, when work is derived, or when inheriting others' work, it is possible to generate another work environment as it is for the work environment of former Job. All the Job names that Place currently opened on the screen points out are the fields displayed as a list, and the job list 89 can change Snap displayed on the snap view 83, if a user chooses one Job with pointing devices, such as a mouse, in this field. In addition, a user is able to perform two or more work in parallel by such selection. If the instruction which deletes Snap which will point out Job chosen by the job list 89 if the DeleteSnap carbon button 90 is pushed from this Place is made and the ImportSnap carbon button 91 is pushed, the instruction which introduces into this Place one related condition Snap chosen in the related condition list window

shown in drawing 43 will be issued.

[0121] These Snap(s) will be added to a set one after another, and the above-mentioned related condition attaching part 65 will hold them, if the set of Snap is held as a related condition and Snap is searched by the related condition extract section 64 from a database 61. Here, since the object of Snap is held in this example, if a user directs one of them with a pointing device in the list display 66, the set of Mediator which the Snap concerned holds will be displayed on the snap view 89. Moreover, processing stored in a database 61 at once when judgment of saving a condition just before being updated by the preservation decision section 68 by the current condition attaching part 65 is made, the Snap is moved to the related condition attaching part 65, the flag which shows the updated thing of "isModifiedFlag" is set and the Save carbon button 75 is pushed so that it may mention later is performed.

[0122] The list display 66 performs a screen display as shown in drawing 43 , and displays in a list the attribute (selection information) of the Snap set currently held on this screen at the related condition attaching part 65. Namely, the selection information of a Snap set is the field shown a list table, and a user can display the list display 101 of the screen concerned on the snap view 83 which showed selected Snap to drawing 42 by choosing any one term of this list with a mouse etc. In addition. In this example, the generated user name, a job name,

and generation time are displayed as selection information of a Snap set. Moreover, if processing which deletes all Snap(s) that the related condition attaching part 65 holds, and are displayed on the list display 66 if the Clear carbon button 102 is pushed is performed and the Close carbon button 103 is pushed, processing which changes the window ( drawing 43 ) concerned into an invisible condition from a user will be performed. In addition, even if it Close(s) in this way, the contents of maintenance of the related condition attaching part 65 are not reset.

[0123] The related condition induction 67 chooses one of Snap(s) which the related condition attaching part 65 holds from the Snap list with which the user was displayed on the list display 66, and is started by pushing the ImportSnap carbon button 91 of the screen of the current status-display section 63. And the related condition induction 67 performs processing made to hold to the current condition attaching part 62 so that selected Snap may be mentioned later, and if , it closes changing the target Snap as a current condition by this. The elements (each Mediator) of a Mediator set which Snap which the current condition attaching part 62 holds points out increase or decrease in number, or the preservation decision section 68 judges whether a condition just before the contents of Snap change is saved taking advantage of saying [ that a user gives directions of reference of Mediator, or a copy ]. In addition, when it should save

by this decision, Snap just before the contents change is saved in a database 61, and restoration of the past work etc. is closed if .

[0124] all Snap(s) that the Place points out when the current condition acquisition section 69 opens Place on a screen -- from a database 61 -- reading -- further -- every -- Job which Snap points out -- reading -- further -- every -- all Mediator(s) that Snap points out are read and processing which makes them generate as an object is performed. Moreover, the current condition acquisition section 69 performs processing which makes this generated object hold to the current condition attaching part 62, and if , it closes carrying out the display output of the target Snap to a display 63 as a current condition by this.

[0125] The related condition extract section 64 extracts the thing corresponding to the selection information specified by a user from from among the data memorized by the database 61, and makes the data concerned hold to the related condition attaching part 65. In order to perform this extract processing, a window screen display as shown in drawing 44 is performed, and, as for the related condition extract section 64, a user performs the retrieval directions for a related condition extract in the screen concerned. Each part of the screen concerned has the following functions. The Place select-list section 110 is a field which displays a Place name in a list as retrieval conditions for a related condition extract, and a user chooses Place of the request under list with

assignment means, such as a mouse.

[0126] The Job select-list section 111 is a field which displays a Job name in a list as retrieval conditions for a related condition extract, and a user chooses Job of the request under list with assignment means, such as a mouse. In addition, in this example, when the CallSnaps carbon button 76 is pushed, only Job which Snap which Place currently opened on the screen has pointed out points out is displayed on a list. What is necessary is here, to give priority to Job of Place opened now, and just to perform displaying on a high order etc., although the number of Job(s) displayed in that case although Job of other Place(s) may also be listed increases very much.

[0127] The User select-list section 112 is a field which displays a User name in a list as retrieval conditions for a related condition extract, and a user chooses User of the request under list with assignment means, such as a mouse. The Mediator name input field 113 are the field which inputs the identifier of Mediator as retrieval conditions for a related condition extract, and a user inputs a desired Mediator name with assignment means, such as a keyboard.

[0128] The Date (From) input field 114 are the field which inputs the beginning of the generation time of Snap as retrieval conditions for a related condition extract, and a user inputs desired time with assignment means, such as a keyboard. The Date (To) input field 115 are the field which inputs the end of the generation time



of Snap as retrieval conditions for a related condition extract, and a user inputs desired time with assignment means, such as a keyboard. Therefore, Snap generated between the time (period) inputted into the Date (From) input field 114 and the Date (To) input field 115 will be searched by the related condition extract section 64.

[0129] If the Retrieve carbon button 116 is pushed, retrieval of Snap will be started by the above-mentioned conditions (selection information) from a database 61, and the window concerned will be closed if the Close carbon button 117 is pushed. If the SharedJobSnaps carbon button 118 is pushed, about Job which Place opened now points out, the same Job will be shared and also processing which searches the newest Snap which points out a user's Job concerned will be performed. In addition, it mentions later in detail about this processing. A push on the MyPastSnaps carbon button 119 performs processing which searches the newest Snap which points out each Job about all Job(s) performed by the Place concerned in the past except Job which Place opened now points out. In addition, it mentions later in detail about this processing.

[0130] Next, actuation of the information processor concerning this example is explained. First, the timing (that is, actuation of the preservation decision section 68) which saves work environment (Snap) in a database 61 is explained. If work is begun as shown in drawing 45 (Job declaration), information gathering will be

performed and various information (Mediator) will be brought together in work environment (Snap). On the other hand, information may be eliminated from work environment, and such a situation can be considered [ like ] to be the timing of a change of work when putting away a desk.

[0131] Then, in case Mediator on Snap starts to decrease from an increment, it performs saving Snap before this change in a database 61, and, thereby, makes it possible to save and reuse the condition of the break of work. Moreover, by adopting the algorithm about this preservation, it can guarantee that no Mediator(s) disappear, and by this, a user cannot stray but can delete now the information which became unnecessary at that time from work environment. Consequently, it becomes unnecessary to think that he will take since it may use later, and the activity burden by the user mitigates.

[0132] With reference to the flow chart which shows the processing about the preservation timing by this preservation decision section 68 to drawing 46 , it explains in more detail. It has three Flag parameters called "isNewFlag" which shows whether each Snap of Place (place) is saved after "alreadyPiledFlag" which shows whether the Snap was already saved, "isModifiedFlag" which shows whether Mediator which the Snap holds had change, and its Snap are newly made, and the preservation decision section 68 has managed these parameters about all Snap(s) currently held at the current condition attaching

part 62 (step S90). In addition, each Snap immediately after opening Place (it having opened) has the Flag parameter in the condition of having called it alreadyPiledFlag=False, isModifiedFlag=False, and isNewFlag=False.

[0133] Subsequently, when it judges whether Mediator was added to Snap (step S91) and Mediator is added to Snap, the Flag parameter of the Snap is changed into alreadyPiledFlag=False and isModifiedFlag=True (step S92). That is, even if it is alreadyPiledFlag=True, it is expressed by adding Mediator that it is not in the already saved condition. Whether on the other hand, when not added, Mediator was deleted judges (step S93), and when deleted, the Flag parameter of the Snap is changed into isModifiedFlag=True (step S94). In addition, alreadyPiledFlag does not change in this case.

[0134] Thus, after deletion of Mediator is performed, it judges whether Snap is saved or not and judges whether "alreadyPiledFlag" is True (step S95). Consequently, since it is necessary to save the condition of Snap just before deleting Mediator to be False while not performing preservation processing since preservation has already accomplished in being True, that Snap is copied (step S96) and copied Snap is stored in a database 61 (step S97). In addition, Snap stored in this database 61 will be held by processing of the related condition extract section 64 at the related condition attaching part 65. Moreover, on the occasion of the above-mentioned copy processing (step S96), the Flag

parameter of Snap just before copying is set as alreadyPiledFlag=True,  
isModifiedFlag=True, and isNewFlag=True.

[0135] Next, it explains with reference to the flow chart which shows the processing in which the preservation decision section 68 stores the updated data in a database 61 to drawing 47 . If the Save carbon button 75 is pushed and a database store instruction is received (step S100), the set of Snap which Place by which current opening is carried out first points out will be gained (step S101). And each snap is turned by the loop formation in order (step S102), and "isModifiedFlag" of each Snap is checked (step S103). Consequently, in being False, while continuing a loop check as it is, in being True, it checks "isNewFlag" further (step S104).

[0136] Consequently, since it is new Snap on the other hand (step S105) when [ for which a database 61 is updated about that Snap ] "isNewFlag" is True when "isNewFlag" is False, that Snap is added to a database 61 (step S106). It carries out repeatedly by the loop formation (step S107), and check and preservation processing are further performed similarly about Place, Job, and Mediator by which current opening is carried out until the check about all Snap(s) finishes a series of above-mentioned processings (step S108). In addition, id of the object when saving performs processing which was mentioned above about the schema of IDMax.

[0137] Next, the retrieval processing which the related condition extract section 64 performs is explained. The case where retrieval directions of Relation Job are taken out with the scene where the need of extracting the work environment of the work which has relation with itself in the situation which is carrying out work may arise, and User nomura has three Job(s) called Job1, Job2, and Job3 in the present condition as the example at drawing 48 is shown. By issuing these directions, other users' newest Snap is searched by the related condition extract section 64 from a database 61 about each Job of Job1, Job2, and Job3. And searched Snap is held at the related condition attaching part 65, and the list is displayed on the list display 66.

[0138] With reference to the flow chart which shows this extract processing to drawing 49, it explains in more detail. first, "listedSnaps" which is the set of Snap which Place (namely, theOpeningPlace) opened now will hold if SharedJobSnaps carbon button 118 carbon button is pushed (step S110) -- referring to -- every -- the Job object of Snap is gained (step S111). And Snap which points out the same Job as them from each gained id of Job is searched (step S112). in addition, the retrieval type used for this retrieval -- for example, SQL etc. -- using -- select\* from Snap where myJob=1 or myJob=2 or myJob=3 and \*\* -- it is expressed like.

[0139] And all the users that appear a retrieval result into reception (step S113)

and a retrieval result in an array etc. are extracted (step S114). In addition, this retrieval result is data like for example, snap[0] = {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, and '{1, 3, 4}'}. Subsequently, a loop formation is turned for every user (step S115), and "createdDate" takes up the newest Snap and makes this each user's newest Snap (step S116). And the related condition attaching part 65 is made to carry out additional storing of (step S117) and all the extracted Snap(s) in the place which ended all the above-mentioned loop-formation processings, and these are displayed on the list display 66 (step S118).

[0140] Next, retrieval processing of the work performed before the related condition extract section 64 carried out is explained. The need of extracting work environment of the work performed before in the situation which is carrying out work (reappearance) may arise, and the case where directions for User nomura to search Job performed before with the scene which has three Job(s) called Job1, Job2, and Job3 in the present condition as the example are taken out to drawing 50 is shown. By issuing these directions, the newest Snap is searched by the related condition extract section 64 from a database 61 about all Job(s) other than Job1 and Job2 from all Snap(s) that he made in the past, and Job3. And searched Snap is held at the related condition attaching part 65, and the list is displayed on the list display 66. Thereby, a user can look through the condition

of having said what kind of other work he would have carried out to the past.

[0141] With reference to the flow chart which shows this retrieval processing to

drawing 51 , it explains in more detail. First, a push on MyPastJobSnap carbon

button 119 carbon button performs retrieval of Snap stored in the database 61

by its user id (this example "1111" of User's nomura id) (step S121). (step S120)

As this retrieval type, the SQL sentence like select \* from Snap where

user=1111 is used, for example. and "listedSnaps" which is the set of Snap

which the related condition extract section 64 extracts all Job(s) that will appear

into a retrieval result if this retrieval result is received (step S122), and Place

(theOpeningPlace) opened now holds further -- referring to -- every -- the Job

object which Snap points out is gained (step S123).

[0142] subsequently, every -- or [ being the same as the job which Snap in Place

which turns a loop formation for every Job (step S124), and those Snap(s) are

opening now points out ] (that is, is it one of Job(s) 1, 2, and 3?) -- \*\* -- what is

said is checked (step S125). Consequently, if the same, while repeating a loop

formation, without doing anything, in not being any of Job 1, 2, and 3, either, the

generation time (createdDate) of Snap uses the newest Snap as each newest

snap of Job for every Job (step S126). And the related condition attaching part

65 is made to carry out additional storing of (step S127) and all the extracted

Snap(s) in the place which ended all the above-mentioned loop-formation

processings, and these are displayed on the list display 66 (step S128).

[0143] Next, introductory processing of the work which the current condition induction 67 performs is explained. The need of sharing the job by the installation to the current condition of a related condition in the situation which is carrying out work may arise, and sharing (that is, collaboration) of Job is realized in this example by introducing Snap from the related condition attaching part 65 to the current condition attaching part 62. In the situation that six Snap(s) are extracted by the related condition attaching part 66, User nomura chooses "Job8 (hayashi) 9 / 18 14:23" as drawing 52 from this inside as that example, the ImportSnap carbon button 91 is pushed, and the case where it is going to introduce into Place (the present condition attaching part 62) to which he is opening this Snap now is shown. Snap which points out Job8 to Place which selected Snap is copied and User nomura is opening will be held, and it means doing Job8 and the job share of Place which User hayashi is opening indirectly by this by performing this installation. It enables this to use the related work environment extract function mentioned above.

[0144] With reference to the flow chart which shows this introductory processing to drawing 53 , it explains in more detail. First, a push on the ImportJob carbon button 91 confirms whether one Snap object is chosen from the related status-display list 101 (step S131). (step S130) Consequently, when not chosen,



while the directions by this carbon button depression are disregarded, when being chosen, the Snap object (theSnap) chosen there is gained (step S132). subsequently, the Job object (theJob) which theSnap points out with reference to "listedSnaps" which is the set of Snap which Place (theOpeningPlace) opened now holds (step S133) -- every of listedSnaps -- it is confirmed whether it is contained in Job which Snap points out (step S134).

[0145] Consequently, when contained, Snap which has pointed out the same job is deleted from listedSnaps of Place, and Snap chosen as the same sequence by the related status-display list is inserted (step S135). And eliminated Snap is added to the related condition attaching part 65, and it is made isModified=True of the snap (step S136). On the other hand, when theJob is not contained, the job to which Snap chosen by the related status-display list points out further Snap chosen by the related status-display list in addition to the head of a snap set (step S137) is added to the head of a job list (step S138).

[0146] Next, the concrete example of use in the 3rd above-mentioned example is explained. In addition, the prototype shown here is programmed by the Java language of Sun Microsystems. Here, the case where User nomura and User hayashi are doing the joint activity is taken for an example, and the example of a screen display when User nomura logs in to Place called "Nomura home" is first shown in drawing 54 . Ten Job(s) called RSP-Practices, Workstory, EvaStory,

Workstory-mk2, CooSS96, Information-Memo, P1G, test, CSCW, and mass media are displayed on the job list 89. It means that this is pointing to Job as which Place called this "Nomura home" holds ten Snap(s) as a current condition to the current condition attaching part 62, and those Snap(s) are displayed on the job list 89, respectively. In this drawing, Job called current and RSP-Practices is chosen and three Mediator(s) are displayed on the snap view 83.

[0147] From this condition, if the CallSnap carbon button 76 is pushed, a screen display as shown in drawing 55 will be made, and the window (refer to drawing 44 ) which gives the retrieval directions for a related condition extract will be started and displayed on the same screen. The list 110 of all Place(s) that can be directed, the list 111 of Job of Place opened now, and the list 112 of all User(s) that can be directed are displayed, and the field 113 which directs the identifier of Mediator and the fields 114 and 115 which specify the period of Date (From) and Date (To) are all over this window. In addition, in this drawing, User hayashi has given the directions which search Snap generated by 15:41 of 15:41 to 9/1 of 9/25.

[0148] And from this condition, if the Retrieve carbon button 116 is pushed, as shown in drawing 56 , a retrieval directions window ( drawing 55 ) will be closed and a retrieval type will be given up to the are recording section 61. And if this

retrieval result comes on the contrary, the window (refer to drawing 43 ) which has the display list of a related condition as shown in drawing 56 will pop up, and it will be displayed on the same screen. In this drawing, "<hayashi> Versioning 9/19 [16:52]" under this list is chosen, and Mediator which that Snap holds is displayed on the snap view 83.

[0149] And when User's hayashi line tends to begin a joint activity about work of "Versioning" which is, Snap chosen within the related status-display list can be added to the present condition, i.e., Place currently opened now, by pushing the ImportSnap carbon button 91. It enables User nomura to do work of "Versioning" with User hayashi by having shown the screen display after performing this in drawing 57 , and carrying out like this. Then, User nomura can know what kind of new information User hayashi added about work of "Versioning" using a What's New function at any time. Furthermore, by taking by processing of the condition composition section 70 so that the sum-set of a Gather Job function, i.e., two persons' work environment, may be mentioned later, the place of common work can be made and the conclusion of final output can be performed in the place to which two persons' work went to some extent.

[0150] The information processor concerning the 4th example of this invention is shown in drawing 58 . In addition, into the same part, the explanation which gives and overlaps the same sign omits the 3rd above-mentioned example. The

condition composition section 70 to which the information processor of this example performs a set operation to the information unit under two or more [ which the current condition attaching part 62 holds in the configuration of the 3rd example ] information-unit sets is added. The condition composition section 70 is a functional means to perform processing which the set operation of the Mediator contained in these Snap(s) is carried out [ processing ] to two or more Snap(s) which the current condition attaching part 62 holds, and makes this result hold to the current condition attaching part 62.

[0151] Moreover, the condition composition section 70 displays a condition composition directions screen window when the Combine carbon button 78 of a display screen (refer to drawing 42 ) is pushed, as shown in drawing 59 , and if the directions which make two or more Snap(s) compound in this window are inputted by the user, it will perform predetermined set operation processing. The carbon button for performing the field for inputting the list section and the various information which display various kinds of information, and various kinds of functions is prepared in the condition composition directions screen window, and the function of these each part is as follows. The current condition list section 120 is a field which displays the list of Job names which all Snap(s) that the current condition attaching part 62 holds point out, and the related condition list section 121 is a field which displays the list of attributes (selection information) of

all Snap(s) that the related condition attaching part 65 holds.

[0152] The base snap carbon button 122 is a carbon button which makes one Snap hold as the base, when calculating difference between two Snap(s). That is, since it is usually single selection, after choosing one Snap, if other Snap(s) are chosen, a selection condition will move. Then, if this carbon button 122 is pushed after choosing one Snap, that selected Snap is used as Base Snap, the function which changes from a degree possible [ multiple selection ] is performed, and a user can change two or more Snap(s) into a selection condition continuously, after pushing this carbon button 122. Thus, Snap used as Base Snap turns into Snap of the side which lengthens when taking difference. In addition, if the or (+) carbon button 123, the subtraction (-) carbon button 124, or the and (\*) carbon button 125 is pushed when no character strings are written to the NewJobName field 127, insertion storing of the Snap of the set operation result by the condition composition section 70 will be carried out by the Job name of Base Snap at the current condition attaching part 62.

[0153] The or (+) carbon button 123 is a carbon button which directs to perform a sum-set operation in the condition composition section 70 about the Mediator set which all Snap(s) by which the multiple selection was made point out by the current condition list and the related condition list. In addition, in case this sum-set is taken, overlapping and not considering as an element is performed,

and Mediator of the same contents is inspecting the identifier of Mediator, and coincidence of URL by this example, in order [ this ] to judge "it is the same." However, it is not necessary to restrict to this inspection approach for example, and it checks to relation with other objects, and identity may be judged or you may check [ \*\*\*\* / checking to a memorandum or time ] to the identity of the version of the document of the reference point.

[0154] the difference of two or more Snap(s) as which, as for the subtraction (-) carbon button 124, Base Snap to others were chosen -- it is the carbon button which directs an operation. That is, the operation  $\text{subtractionSnap} = \text{BaseSnap} - (\text{Snap1} + \text{Snap2} + \dots)$  is made to perform in the condition composition section 70.

The and (\*) carbon button 125 is a carbon button which directs to perform a common operation in the condition composition section 70 about the Mediator set which all Snap(s) by which the multiple selection was made point out by the current condition list and the related condition list. The Close carbon button 126 is a carbon button which performs the directions which make the condition composition directions screen concerned closed.

[0155] The NewJobName field 127 makes Job of Snap which is that result of an operation in case the current condition attaching part 62 is made to carry out insertion maintenance of the set operation result the job name of the character string written to the field 127 concerned, and when this field 127 is empty, the

result of an operation becomes the job name of Base Snap. The GatherJob carbon button 128 is a carbon button which calls a job result intensive function, and mentions this detail later. The What'sNew carbon button 129 is a carbon button which calls a What's New function, and mentions this detail later. [0156] The JobEssence carbon button 130 is a carbon button which calls a Job Essence function, and this function takes and displays the common class of each one of newest Snap(s) about that job, when a job is shared by two or more persons and work is being done. It can check whether the share of the indispensable information, such as missions currently shared, reference reference, and a presentation place, is carried out exactly by this. Moreover, the member which newly joins a job can draw a place from share information, and can prepare an environment.

[0157] Next, set operation processing of the information processor concerning this example is explained. First, a sum-set operation (job result intensive function) is explained with reference to the example of condition composition for result concentration of the job shown in drawing 60 . In this example, supposing it takes out job result intensive directions with the scene where User nomura has three Job(s), Job1, Job2, and Job3, in the current condition, other users' newest Snap will be searched with it from a database 61 about each Job of Job1, Job2, and Job3. Two persons' newest, sekijima and hayashi, Snap is searched with

this example for three persons' newest, sekijima, hayashi, and hazama, Snap about Job2 about Job1. And the sum-set operation of Snap searched also including Snap of nomura for every Job is performed, and the result is inserted in the present condition attaching part 62 by the job name of Job1 (+) and Job2 (+).

[0158] With reference to the flow chart which shows this job result intensive function (sum-set data processing) to drawing 61 , it explains in more detail. first, listedSnaps which is the set of Snap which Place (theOpeningPlace) opened now will hold if the GatherJob carbon button 128 is pushed (step S140) -- referring to -- every -- the Job object of Snap is gained (step S141). And Snap which points out the same Job as them from each gained id of Job is searched with an SQL sentence called select \* from Snap where myJob=1 or myJob=2 or myJob=3 (step S142).

[0159] And all User(s) that appear this retrieval result into reception, (step S143), and this retrieval result in the array for example, snap[0] = {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, and '{1, 3, 4}'} are extracted (step S144). And a loop formation is turned for every User (step S145), and createDate makes the newest Snap each user's newest Snap (step S146). Subsequently, only each user's number of Newest Snap turns a loop formation (step S147), and only the number of Mediator(s) which each Snap holds further turns a loop formation (step S148). In it, further, only the number of all Mediator(s) already added to



Composition Snap turns a loop formation (step S149), it confirms whether new Mediator is the same as already added Mediator (step S150), and in not being the same, it adds new Mediator to Composition Snap (step S151). It is repeatedly performed until it stops grinding such processing Mediator with Snap which carries out a loop formation (step S152).

[0160] next, difference -- about data processing (What's New function), as shown in drawing 62 , its line is explained with reference to the example of condition composition for extracting the new information about Job which is (namely, What's New function). Suppose that the What'sNew carbon button 129 was pushed and the What's New instruction was taken out with the scene where User nomura has three Job(s), Job1, Job2, and Job3, in the current condition in this example. In addition, Snap which the user nomura holds is shown by this drawing as A1, B1, and C1. A thing newer than User's nomura Snap is extracted from a database 61 by other users' newest Snap about each Job of Job1, Job2, and Job3 as a result of this instruction. That is, about Job1, A2 [ newer than A1 ] of User sekijima, B-2, and User's hayashi A4 are extracted.

[0161] Information to extract by the What's New function here For example, since it is the information to which other users generated [ User nomura ] A1 about Job1 after creation It is related with A2 of User sekijima extracted as new Snap, B-2, and User's hayashi A4. The Snap [ of the user made just before

User's nomura Snap ] of Job can be extracted ( drawing 62 A3, B4, A5), and it can know what the new information which User nomura is not looking at is by taking difference with these. that is, User's nomura new information can be acquired by formula (however -- here -- "+" -- a sum-set composition operation and "-" -- difference -- it is shown that a synthetic operation and sigma take the sum-set of the result performed to all users that become a candidate for a comparison.) called new information =sigma(User's X new Snap - just before [ Snap ] User X)-nomuraSnap. In addition, in the example of drawing 62 , count, such as Job1 (new)=(A2-A3)+(A4-A5)-A1 and Job2 (new)=(B-2-B4)-B1, will be performed.

[0162] With reference to the flow chart which shows this What's New function (difference set operation processing) to drawing 63 , it explains in more detail. First, if the What'sNew carbon button 129 is pushed (step S160), with reference to listedSnaps which is the set of Snap which Place (theOpeningPlace) opened now holds, the origin Snap of some will be gained out of this (step S161), and the generation time of these origin Snap will be acquired (step S162). Subsequently, Snap which points out the same Job as them from each gained id of Job is searched with an SQL sentence called select \* from Snap where myJob=1 or myJob=2 or myJob=3 (step S163).

[0163] This retrieval result in the array for example, snap[0] = {1, 'Wed Jul 10

15:00:00 JST 1996', 1111, 1, 0, and '{1, 3, 4}'} And reception (step S164), All  
Usre(s) that appear into this retrieval result are extracted (step S165). every --  
every User -- a loop formation -- turning (step S166) -- every of a retrieval result  
-- createDate of Snap The Snap (A2 of drawing 62 , B-2, A4) newest by the Job  
of the user is extracted as each user's newest snap more newly than  
createDate of the origin Snap which points out the same Job (step S167).  
Subsequently, the newest thing (A3 of drawing 62 , B4, A5) is extracted as just  
before [ Snap ] each user by Snap made before createDate of the origin Snap  
which is the same user as each extracted newest Snap, or the same job, and  
points out the same Job (step S168).

[0164] And Newest Snap and the difference of a just before Snap are taken for  
every Job, and the sum-set is taken (step S169). furthermore, every -- the  
difference of Sum-set Snap is taken from the origin Snap of Job, and Job of the  
identifier of "a job name (new)" for it is made as the newcomer Mediator set of  
Job (step S170). subsequently, Job (that is, "job name (new)") of Sum-set Snap  
-- every of listedSnaps -- whether it is already contained in Job which Snap  
points out judges (step S171), when contained, Snap which has pointed out the  
same Job is deleted from listedSnaps, and Snap chosen as the same sequence  
by the related status-display list is inserted (step S172). Furthermore, eliminated  
Snap is added to the related condition attaching part 65, and it is made

isModified=True about the Snap (step S173). On the other hand, when not contained, Job to which Sum-set Snap points out Sum-set Snap in addition to the head of a Snap set (step S174) is added to the head of a Job list (step S175). Such processing is repeatedly performed until User which carries out a loop formation is lost (step S176).

[0165] Next, the concrete example of use in the 4th above-mentioned example is explained. In addition, the prototype shown here is programmed by the Java language of Sun Microsystems. Here, the case where User nomura and User hayashi are doing the joint activity is taken for an example, first, to drawing 64 , User nomura is opening Place63 "Nomura home", and the condition of opening the window 140 of a related status-display list and the condition composition directions window (referring to drawing 59 ) is shown further. In addition, "<hayashi>P1G 9/10[15:51]" which is related status-display listing and is chosen is displayed on the snap view 83.

[0166] this -- a condition -- oneself -- carrying out -- \*\*\*\* -- work -- it is -- "--  
CooSS -- 96 -- " -- a user -- hayashi -- the past -- a place -- a condition -- it is -- "--  
< -- hayashi>P -- one -- G -- nine -- /-- ten -- [-- 15:51 --] -- " -- doubling -- being  
new -- work -- it was going to begin -- \*\* -- carrying out . In this case, first,  
"CooSS96" under list 120 (that is, current condition list) of Hidari in a condition  
composition directions window is chosen, and after driving the base snap carbon

button 122 into the NewJobName field 127 with "New Project" further by choosing "<hayashi>P1G 9/10[15:51]" from push, then the list 121 (that is, related condition list) of right, the GatherJob carbon button 128 is pushed.

[0167] Consequently, the sum-set operation of Snap which has pointed out the Job "CooSS96" which the present condition holds, and Snap of "<hayashi>P1G 9/10[15:51]" which a related condition holds is performed, and as a result, as shown in drawing 65 , the new Job "New Project" is generated, and additional storing is carried out at the present condition attaching part 62, and it is inserted in the present condition (that is, Place currently opened now). Moreover, all Mediator(s) that two Snap(s) held are displayed on the snap view 83. However, when the Mediator same at this time as Snap which performs a sum-set is contained, that Mediator overlaps and is not copied.

[0168]

[Effect of the Invention] While according to the information processor of this invention being able to hold information required for a certain business with an information-unit maintenance means and being able to carry out business using the work environment so that clearly from the above explanation, by directing selection information, the information-unit set of the request which an information-unit set maintenance hysteresis storage means memorizes can be called, the condition of the place of work can be changed, and another work can

be resumed from there. Moreover, even if it does not classify beforehand, also after desired work environment is acquired and work progresses, work environment can be called with various views, such as an identifier of work, time, a user name, and an information-unit name. Moreover, since the condition of the past work is held, information required for future work can be extracted from there returning to the condition at the time of the past and starting work, and by restoring the past condition.

[0169] Moreover, in order for a selection information grant means to give selection information, such as time, a user name, and a work name, to an information-unit set, to memorize for an information-unit set maintenance hysteresis storage means and to indicate the information-unit set by classification by selection information at an information-unit set maintenance hysteresis classification display means, to a user, it is easy to understand and the change of work environment can be shown. Moreover, business can be advanced on many sides by giving two or more selection information to the information-unit set, and changing the provisions of classification in an information-unit set maintenance hysteresis classification display means according to a demand of a user. Moreover, by saving two or more conditions of the place of work of performing one task, the condition and related information of the past work can be referred to, or it can return to the condition of the past work

and new work can be begun.

[0170] Moreover, since the hysteresis of the condition of the place of work is accumulated, some [ or ] information which a certain work is sufficient as some information required for a certain work, and is used can extract and reuse the know-how of work of something in the information on a proper at a certain work. Moreover, a user can build a new environment by extracting required information from the condition of the place of the past work. Moreover, a user can build a new environment by compounding the condition of the place of work of some past.

[0171] Furthermore, the activity progress and the result which last from themselves but others' past now are specifically utilized, an environment required for new work is generated, it becomes possible to begin work there, and efficient work can be realized. [ not only ] Moreover, the framework which is easy to exchange each other knowledge between loose coordination members (community) can be offered. Moreover, in an information share into a group, or a status control, not the collaboration in single business like before but action of work of others, a situation, and synthetic collaboration that can reflect progress can be made possible. Moreover, except the job which is in the current line, since the framework with easy restoring this is offered even if it deletes from a current condition, the efficiency of work by the user improves by leaps and

bounds.

---

## DESCRIPTION OF DRAWINGS

---

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram of the information processor concerning the 1st example of this invention.

[Drawing 2] It is a conceptual diagram explaining the relation between an information-unit set and selection information.

[Drawing 3] It is a conceptual diagram explaining storage and read-out of an information-unit set for an information-unit set maintenance hysteresis storage means.

[Drawing 4] It is a conceptual diagram explaining manifestation of the class of work, and the change of work.

[Drawing 5] It is drawing showing an example of a display of an information-unit set and a selection information list.

[Drawing 6] It is drawing showing other examples of a display of an information-unit set and a selection information list.

[Drawing 7] It is drawing showing an example of further others of the display of



an information-unit set and a selection information list.

[Drawing 8] It is the explanatory view showing an example of the DS about a user.

[Drawing 9] It is the explanatory view showing an example of the DS about a workspace.

[Drawing 10] It is the explanatory view showing an example of the DS about a job.

[Drawing 11] It is the explanatory view showing an example of the DS corresponding to a workspace-job.

[Drawing 12] It is the explanatory view showing an example of the DS about an information-unit set.

[Drawing 13] It is the explanatory view showing an example of the DS about an information unit.

[Drawing 14] It is the explanatory view showing an example of the DS corresponding to a workspace-information-unit set.

[Drawing 15] It is the explanatory view showing an example of the DS corresponding to an information-unit set-information unit.

[Drawing 16] It is the explanatory view showing an example of the DS about selection information.

[Drawing 17] It is the explanatory view showing an example of the DS about a

selection information list.

[Drawing 18] It is the explanatory view showing an example of the accumulated series of commands.

[Drawing 19] It is the explanatory view showing an example of the storage activation Ruhr.

[Drawing 20] It is the flow chart which shows an example of the procedure of information-unit incorporation processing.

[Drawing 21] It is the flow chart which shows an example of the procedure of storage decision processing of an information-unit set.

[Drawing 22] It is the flow chart which shows an example of the procedure of new-job registration processing.

[Drawing 23] It is the flow chart which shows an example of the procedure of the classification display process of selection information.

[Drawing 24] It is the flow chart which shows an example of the procedure of reading processing of an information-unit set.

[Drawing 25] It is drawing showing an example of a display of an information-unit set and a selection information list.

[Drawing 26] It is drawing showing an example of a display of an information-unit set and a selection information list.

[Drawing 27] It is drawing showing an example of a display of an information-unit

set and a selection information list.

[Drawing 28] It is drawing showing an example of a display of an information-unit

set and a selection information list.

[Drawing 29] It is drawing showing an example of a display of an information-unit

set and a selection information list.

[Drawing 30] It is the block diagram of the information processor concerning the

2nd example of this invention.

[Drawing 31] It is drawing explaining synthetic directions of two or more

information-unit sets.

[Drawing 32] It is drawing explaining selection directions of the information-unit

set to compound.

[Drawing 33] It is drawing explaining filtering directions.

[Drawing 34] It is a conceptual diagram explaining synthetic processing of two or

more information-unit sets.

[Drawing 35] It is the flow chart which shows an example of the procedure of

sum-set data processing.

[Drawing 36] It is the flow chart which shows an example of the procedure of

common-class data processing.

[Drawing 37] difference -- it is the flow chart which shows an example of the

procedure of data processing.

[Drawing 38] It is the flow chart which shows an example of the procedure of filtering processing.

[Drawing 39] It is the block diagram of the information processor concerning the 3rd example of this invention.

[Drawing 40] It is drawing explaining the schema of a database.

[Drawing 41] It is drawing explaining DS.

[Drawing 42] It is drawing showing the example of a display of the Main screen.

[Drawing 43] It is drawing showing an example of the display list screen of the snap held as a related condition.

[Drawing 44] It is drawing showing an example of the retrieval directions screen for a related condition extract.

[Drawing 45] It is drawing explaining the timing of work environment preservation.

[Drawing 46] It is the flow chart which shows the procedure of preservation timing processing.

[Drawing 47] It is the flow chart which shows the procedure of the storing processing to a database.

[Drawing 48] It is drawing explaining the retrieval processing for the work environment extract of the work which has relation with itself.

[Drawing 49] It is the flow chart which shows the procedure of related environmental extract processing.

[Drawing 50] It is drawing explaining the retrieval processing for the work environment extract of the work which he performed before.

[Drawing 51] It is the flow chart which shows the procedure of environmental extract processing of work before itself.

[Drawing 52] It is drawing explaining the introductory processing to the current condition of a related condition.

[Drawing 53] It is the flow chart which shows the procedure of the job share processing by installation.

[Drawing 54] It is drawing showing the example of a display of the Maine screen.

[Drawing 55] It is drawing showing an example for the already saved retrieval directions screen of Snap.

[Drawing 56] It is drawing showing the example of a screen display which one of the retrieval results was directed [ example ] and displayed the snap on the snap view.

[Drawing 57] When a snap is introduced into a current condition attaching part, it is drawing showing the example of a screen display.

[Drawing 58] It is the block diagram of the information processor concerning the 4th example of this invention.

[Drawing 59] It is drawing showing an example of the screen of condition composition directions.

[Drawing 60] It is drawing explaining the condition composition for result concentration of a job.

[Drawing 61] It is the flow chart which shows the procedure of a job result intensive function.

[Drawing 62] Its line is drawing explaining the condition composition which extracts the new information about the job which is.

[Drawing 63] It is the flow chart which shows the procedure of a What's New function.

[Drawing 64] It is drawing showing the screen display which inputs synthetic directions of two snaps.

[Drawing 65] It is drawing showing the screen which displayed the synthetic result of an operation of two snaps.

[Description of Notations]

1 ... Server 2 ... The information-unit storage section, 3 ... Information-unit set maintenance hysteresis storage section, [ 6 ... Selection information grant section, ] 4 ... The information-unit set selection section, 5 ... Information-unit set read-out section 7 ... Storage directions section 8 ... The actuation command are recording section, 9 ... Information-unit set storage decision section, [ 12 ... I/O section, ] 10 ... A network, 11 ... Client 13 ... Information-unit set display 14 ... Information-unit set attaching part, 15 ... Information-unit editorial department

16 ... Selection information list display, 17 ... Selection information directions  
section 61 ... Are recording section (database), 62 ... Current condition attaching  
part 63 [ 65 / 67 / 69 ... The current condition acquisition section, 70 ... Condition  
composition section, / ... Related condition induction, 68 ... Preservation decision  
section / ... A related condition attaching part 66 ... List display ] ... The current  
status-display section, 64 ... Related condition extract section

---

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-31660

(43) 公開日 平成10年(1998) 2月3日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/00			G 0 6 F 15/20	Z
3/14	3 4 0		3/14	3 4 0 A
12/00	5 1 5		12/00	5 1 5 A

審査請求 未請求 請求項の数37 F D (全 50 頁)

(21) 出願番号 特願平8-302511

(22) 出願日 平成8年(1996)10月28日

(31) 優先権主張番号 特願平8-148366

(32) 優先日 平8(1996)5月17日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂二丁目17番22号

(72) 発明者 野村 恭彦

神奈川県足柄上郡中井町境430 グリーン

テクなかい 富士ゼロックス株式会社内

(72) 発明者 林 浩一

神奈川県足柄上郡中井町境430 グリーン

テクなかい 富士ゼロックス株式会社内

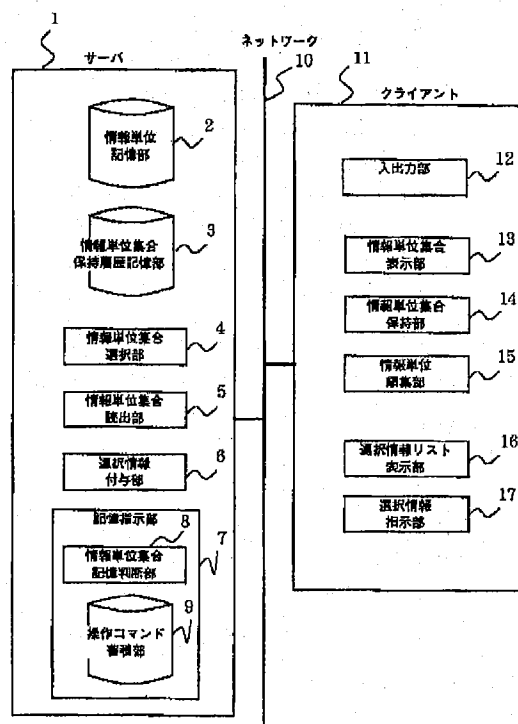
(74) 代理人 弁理士 守山 辰雄

(54) 【発明の名称】 情報処理装置

(57) 【要約】

【課題】 作業環境において行なわれた活動を反映し、活動に必要な情報を保持する作業環境を自動的に構築し、業務毎に情報を整理する情報処理装置を提供する。

【解決手段】 作業環境を情報単位集合 (Snap) とし、作業環境で利用されるデータオブジェクト等を情報単位 (Mediator) とし、情報単位集合保持履歴記憶部3に複数の情報単位集合をPlaceでまとめて記憶する一方、情報単位集合保持部14に作業対象となる情報単位集合を保持させる。仕事の変更等といった所定の条件が整うと、記憶指示部7により情報単位集合保持部14が保持する情報単位集合を情報単位集合保持履歴記憶部3に記憶せしめ、この情報単位集合に選択情報付与部6が仕事名等といった選択情報を付与する。或る情報単位集合を作業対象とするために利用者から選択情報が入力されると、情報単位集合選択部4がこれを用いて、記憶部3から該当する情報単位集合を選択し、情報単位集合読出部5によって情報単位集合保持部14へ読み出される。





**【特許請求の範囲】**

**【請求項1】** データオブジェクトと前記データオブジェクトを操作するためのアプリケーションオブジェクトとを含んで構成され、前記データオブジェクトを前記アプリケーションオブジェクトを用いて収集および作成するための作業空間を備える情報処理装置において、データオブジェクト又はアプリケーションオブジェクトである情報単位の集合を保持する情報単位集合保持手段と、前記情報単位集合を複数記憶することが可能な情報単位集合保持履歴記憶手段と、前記情報単位集合保持手段が保持する前記情報単位集合を前記情報単位集合保持履歴記憶手段に記憶せしめる記憶指示手段と、前記記憶指示手段により前記情報単位集合保持履歴記憶手段へ記憶される前記情報単位集合に選択情報を付与する選択情報付与手段と、前記情報単位集合保持手段に記憶させる前記情報単位集合を前記選択情報付与手段により付与された前記選択情報を用いて前記情報単位集合保持履歴記憶手段に記憶された前記情報単位集合の中から選択する選択手段と、前記選択手段によって選択された前記情報単位集合を前記情報単位集合保持手段へ読み出す読出手段と、を備えたことを特徴とする情報処理装置。

**【請求項2】** 請求項1に記載の情報処理装置において、前記情報単位集合保持手段が保持する前記情報単位集合に対して情報単位を編集する情報単位編集手段と、前記情報単位編集手段により追加或いは削除された前記情報単位名を記憶する増減情報単位名記憶手段と、を更に備え、前記選択情報付与手段は前記選択情報として前記増減情報単位名記憶手段が記憶する前記情報単位名を選択情報として前記情報単位集合に対して付与し、前記選択手段は前記選択情報付与手段により付与された前記情報単位名を用いて前記情報単位集合を選択することを特徴とする情報処理装置。

**【請求項3】** 請求項1又は請求項2に記載の情報処理装置において、前記情報単位集合保持手段が保持する情報単位を指示する情報単位指示手段と、前記情報単位指示手段により指示された前記情報単位を移動或いは複写する先となる前記情報単位集合保持履歴記憶手段内の前記情報単位集合を選択する第2の選択手段と、前記情報単位指示手段により指示された前記情報単位を前記第2の選択手段により選択された前記情報単位集合に追加する情報単位移動手段と、前記情報単位移動手段により前記情報単位を情報単位集合間で移動させることに基づいて移動元と移動先の各々

の前記情報単位集合名を記憶する移動元情報単位集合名記憶手段と、を更に備え、前記選択情報付与手段は前記選択情報として前記移動元情報単位集合名記憶手段が記憶する移動元の情報単位集合名を格納し、前記選択手段は前記選択情報付与手段により付与された前記移動元情報単位集合名を用いて前記情報単位集合を選択することを特徴とする情報処理装置。

**【請求項4】** 請求項1乃至請求項3のいずれか1項に記載の情報処理装置において、前記情報単位集合保持手段が保持する前記情報単位に対して処理対象とする指示選択を受け付ける情報単位指示手段と、前記情報単位指示手段により指示された情報単位に対して処理を行う情報単位処理手段と、前記情報単位処理手段により処理が実行された情報単位名を記憶する実行情報単位名記憶手段と、を更に備え、前記選択情報付与手段は前記選択情報として前記実行情報単位名記憶手段が記憶する前記情報単位名を選択情報として前記情報単位集合に対して付与し、前記選択手段は前記選択情報付与手段により付与された前記情報単位名を用いて前記情報単位集合を選択することを特徴とする情報処理装置。

**【請求項5】** 請求項4に記載の情報処理装置において、前記情報単位処理手段により前記情報単位に対して実行されたコマンド名を記憶する実行コマンド名記憶手段を更に備え、前記選択情報付与手段は前記選択情報として前記実行コマンド名記憶手段が記憶する前記コマンド名を選択情報として前記情報単位集合に対して付与し、前記選択手段は、前記選択情報付与手段により付与された前記コマンド名を用いて前記情報単位集合を選択することを特徴とする情報処理装置。

**【請求項6】** 請求項1乃至請求項5のいずれか1項に記載の情報処理装置において、前記情報単位集合保持履歴記憶手段に記憶されている前記情報単位集合に付与された前記選択情報を分類表示する選択情報分類表示手段と、前記選択情報分類表示手段により表示される選択情報の表示形式を保持する選択情報分類表示形式保持手段と、前記選択情報分類表示手段により表示された前記選択情報を指示する選択情報指示手段と、を更に備え、前記情報単位集合保持履歴記憶手段は前記選択情報を1以上の選択情報属性ごとに分けて前記情報単位集合に付与して記憶し、前記選択情報分類表示手段は第1階層の選択情報属性表示領域と第2階層の選択情報属性表示領域とを有し、前記第1階層の選択情報属性表示領域には第1階層として前記選択情報分類表示形式保持手段により定められた形

式で前記選択情報属性の選択情報をリスト表示し、前記第2階層の選択情報属性表示領域には第2階層として前記選択情報分類表示形式保持手段により定められた形式で前記選択情報属性の選択情報をリスト表示し、前記選択情報指示手段により前記第1階層の前記選択情報が指示されたことに基づいて当該指示された選択情報を含む前記情報単位集合保持履歴記憶手段に記憶されている情報単位集合に付与された第2階層の選択情報のみを前記第2階層の選択情報属性表示領域に表示し、前記選択手段は前記1階層で指示された選択情報と前記第2階層で指示された選択情報の双方を用いて前記情報単位集合保持履歴記憶手段が記憶する前記情報単位集合の1つを選択することを特徴とする情報処理装置。

【請求項7】 請求項6に記載の情報処理装置において、前記情報単位集合保持履歴記憶手段が記憶する前記情報単位集合に付与されている全ての前記選択情報に対して各選択情報の付与されている最新の前記情報単位集合の作成日時を保持する選択情報最新日時保持手段を更に備え、前記選択情報分類表示手段は前記選択情報を前記選択情報最新日時保持手段が保持する前記最新の作成日時により並べ替えて表示することを特徴とする情報処理装置。

【請求項8】 請求項6に記載の情報処理装置において、前記情報単位集合保持履歴記憶手段が記憶する前記情報単位集合に付与されている全ての前記選択情報に対して全ての前記情報単位集合の作成日時に対応して付与された日時を保持する選択情報付与日時保持手段を更に備え、前記選択情報分類表示手段は前記選択情報を前記選択情報付与日時保持手段が保持する日時に対応して並べ替えて表示することを特徴とする情報処理装置。

【請求項9】 請求項6乃至請求項8のいずれか1項に記載の情報処理装置において、前記選択情報分類表示手段は前記情報単位集合保持履歴記憶手段に記憶される前記情報単位集合に付与された前記選択情報を表示し、前記選択手段は前記選択情報分類表示手段により表示された前記選択情報の内の前記情報単位集合保持手段へ読み込む前記情報単位集合に付与された選択情報の指示を受け付けて指示された選択情報を有する情報単位集合の内で最新のものをを選択することを特徴とする情報処理装置。

【請求項10】 請求項6乃至請求項8のいずれか1項に記載の情報処理装置において、前記記憶指示手段により前記情報単位集合保持履歴記憶手段に前記情報単位集合の記憶を指示することに基づいて前記情報単位集合保持履歴記憶手段に記憶する前記情報単位集合に付与する選択情報を利用者により入力或いは指示させる選択情報入力手段を更に備え、

前記選択情報入力手段は前記情報単位集合の格納時に利用者が行っていた仕事名の入力を受け付け、前記選択情報分類表示手段は前記情報単位集合保持履歴記憶手段に記憶する前記仕事名を表示或いは前記仕事名によって前記情報単位集合を分類表示することを特徴とする情報処理装置。

【請求項11】 請求項6に記載の情報処理装置において、前記選択情報分類表示手段は前記情報単位集合保持履歴記憶手段に記憶する利用者名を表示或いは利用者名によって前記情報単位集合を分類表示することを特徴とする情報処理装置。

【請求項12】 請求項6に記載の情報処理装置において、前記選択情報分類表示手段は前記情報単位集合保持履歴記憶手段に記憶する前記情報単位名を表示或いは前記情報単位名によって情報単位集合名を分類表示することを特徴とする情報処理装置。

【請求項13】 請求項1乃至請求項12のいずれか1項に記載の情報処理装置において、前記情報単位集合保持手段或いは前記情報単位集合保持履歴記憶手段に保持された2以上の前記情報単位集合を選択する第3の選択手段と、前記第3の選択手段によって選択された2以上の情報単位集合が保持する前記情報単位の集合演算を行う情報単位集合合成手段を更に備え、前記集合演算を行って作成された新たな情報単位集合を前記情報単位集合保持手段に読み込み、前記情報単位集合保持手段の保持する前記情報単位集合とすることを特徴とする情報処理装置。

【請求項14】 請求項13に記載の情報処理装置において、前記情報単位集合保持手段或いは前記情報単位集合保持履歴記憶手段に保持された前記情報単位集合を選択する第4の選択手段と、前記第4の選択手段によって選択された情報単位集合が保持する情報単位の集合に対して情報単位の絞り込み条件を指示する絞り込み条件指示手段と、前記第4の選択手段によって選択された前記情報単位集合から前記絞り込み条件指示手段により指示された絞り込み条件を満たしていないものを削除する絞り込み実行手段と、を更に備え、前記情報単位集合に対して前記情報単位の絞り込みを行って新たな情報単位集合を作成し、当該新たな情報単位集合を前記情報単位集合保持手段に読み込み、前記情報単位集合保持手段が保持する情報単位集合とすることを特徴とする情報処理装置。

【請求項15】 請求項1乃至請求項14のいずれか1項に記載の情報処理装置において、利用者或いはコマンドからの指示によるイベントが発生

したことに基づいて前記情報単位集合保持履歴記憶手段に情報単位集合を格納するか否かを予め定められたルールに基づいて判定する記憶実行判断手段を更に備え、前記情報単位編集手段は前記記憶実行判断手段による判断に基づいて前記情報単位集合保持手段が保持する情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項16】 請求項15に記載の情報処理装置において、記憶実行判断手段は前記情報単位集合保持履歴記憶手段が記憶する情報単位集合を前記情報単位集合保持手段に読み出すイベントが発生したことに基づいて前記判断を行い、前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項17】 請求項15に記載の情報処理装置において、利用者からの仕事の種類の変化の表明を受け付ける変化表明手段を更に備え、前記記憶実行判断手段は前記変化表明手段からの表明イベントが発生したことに基づいて前記判断を行い、前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項18】 請求項15に記載の情報処理装置において、利用者を一意に判別する利用者管理手段を更に備え、前記記憶実行判断手段は前記利用者管理手段が利用者の変化を判別したイベントが発生したことに基づいて前記判断を行い、前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項19】 請求項15に記載の情報処理装置において、前記情報単位集合保持手段が保持する情報単位集合の情報単位の追加或いは削除或いは属性変更を検出する情報単位変更検出手段を更に備え、前記記憶実行判断手段は前記情報単位変更検出手段が前記変更を検出したイベントが発生したことに基づいて前記判断を行い、前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項20】 請求項15に記載の情報処理装置にお

いて、  
処理対象となる情報単位の内容を表示させたことを検出する情報単位表示検出手段を更に備え、  
前記記憶実行判断手段は前記情報単位表示検出手段が前記表示を検出したイベントが発生したことに基づいて前記判断を行い、  
前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項21】 請求項15に記載の情報処理装置において、前記情報単位集合保持履歴記憶手段に情報単位集合を記憶させる周期を保持する記憶実行周期保持手段を更に備え、前記記憶実行判断手段は前記周期毎のイベントに基づいて前記判断を行い、前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項22】 請求項15に記載の情報処理装置において、コマンドが実行された回数に基づく規則を保持する記憶実行規則保持手段と、実行された1以上のコマンドを記憶しておくコマンド記憶手段と、を更に備え、前記記憶実行判断手段は予め定められたコマンドが前記記憶実行規則保持手段が保持する規則の順序と一致したイベントに基づいて前記判断を行い、前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項23】 請求項15に記載の情報処理装置において、前記情報単位移動手段により情報単位が前記情報単位集合保持手段から前記情報単位集合保持履歴記憶手段へ移動されたことを検出する情報単位移動検出手段を更に備え、

前記記憶実行判断手段は前記情報単位移動検出手段が前記移動を検出したイベントが発生したことに基づいて前記判断を行い、  
前記情報単位編集手段は当該イベント発生時に前記情報単位集合保持手段が保持している情報単位集合を前記情報単位集合保持履歴記憶手段に格納することを特徴とする情報処理装置。

【請求項24】 請求項1に記載の情報処理装置において、前記情報単位集合保持手段は、表示対象となる複数の情

報単位集合をその属性としての選択情報と共に保持する現在状態保持手段と、当該現在状態保持手段に保持された情報単位集合の内の利用者により入力された選択情報に対応する情報単位集合を表示出力する現在状態表示手段と、を備え、

前記記憶指示手段は、情報単位集合中の情報単位の数が増加された場合に当該変更前の情報単位集合を保存するか否かを予め設定した規則に従って判断する保存判断手段と、当該保存判断手段により保存すると判断された情報単位集合を前記情報単位集合保持履歴記憶手段へ記憶させる蓄積手段と、を備えていることを特徴とする情報処理装置。

【請求項 25】 請求項 24 に記載の情報処理装置において、

前記現在状態表示手段は、前記現在状態保持手段が保持する全ての情報単位集合を識別するための情報を表示出力することを特徴とする情報処理装置。

【請求項 26】 請求項 24 又は請求項 25 に記載の情報処理装置において、

前記情報単位集合保持履歴記憶手段は、複数の情報単位集合を更に集合化したプレースを複数記憶しており、これらプレースの内の利用者により選択されたプレースを前記情報単位集合保持手段に保持させる現在状態獲得手段を更に備えていることを特徴とする情報処理装置。

【請求項 27】 請求項 24 又は請求項 25 に記載の情報処理装置において、

前記情報単位集合保持履歴記憶手段は、複数の情報単位集合を更に集合化したプレースを複数記憶しており、予め設定した条件に基づいてこれらプレースの内のいずれかのプレースを前記情報単位集合保持手段に保持させる現在状態獲得手段を更に備えていることを特徴とする情報処理装置。

【請求項 28】 請求項 24 に記載の情報処理装置において、

前記保存判断手段は、情報単位集合中の情報単位の数が増加から減少に転じた際に保存すべきものと判断することを特徴とする情報処理装置。

【請求項 29】 請求項 24 に記載の情報処理装置において、

前記選択手段は、予め設定された選択情報についての抽出規則に対応する情報単位集合を前記情報単位集合保持履歴記憶手段に保持された情報単位集合の中から抽出する関連状態抽出手段と、当該抽出された情報単位集合を保持する関連状態保持手段と、当該抽出された情報単位集合の選択情報を利用者に対して表示するリスト表示手段と、表示された選択情報に対する利用者の要求に対応する情報単位集合を前記関連状態保持手段に保持されている情報単位集合中から指定する関連情報指定手段と、を備え、

前記読出手段は、前記関連情報指定手段により指定され

た情報単位集合を前記関連状態保持手段から読み出して前記現在状態保持手段に保持させる関連状態導入手段を備えていることを特徴とする情報処理装置。

【請求項 30】 請求項 29 に記載の情報処理装置において、

前記現在状態保持手段は選択情報として仕事名を有しており、

前記関連状態導入手段は、前記指定された情報単位集合が保持する情報単位集合と仕事名を複写して前記現在状態保持手段に保持させ、

前記関連状態抽出手段は、前記現在状態保持手段が保持する情報単位集合の仕事名と同じ仕事名を有する 1 以上の関連する情報単位集合を抽出することを特徴とする情報処理装置。

【請求項 31】 請求項 30 に記載の情報処理装置において、

前記関連状態抽出手段は、前記現在状態保持手段が保持する情報単位集合の選択情報である仕事名と同じ仕事名を有する他の利用者の生成した情報単位集合の内で利用者毎に最新の情報単位集合を抽出することを特徴とする情報処理装置。

【請求項 32】 請求項 30 に記載の情報処理装置において、

前記関連状態抽出手段は、利用者自身がそれまでに生成した全ての情報単位集合の内でその選択情報である仕事名毎に最新の情報単位集合を前記情報単位集合保持履歴記憶手段から抽出することを特徴とする情報処理装置。

【請求項 33】 請求項 24 に記載の情報処理装置において、

前記現在状態保持手段が保持する 2 以上の情報単位集合の要素である情報単位に対して予め設定した集合演算を行う状態合成手段を、更に備え、

当該状態合成手段は当該集合演算を行って合成された情報単位集合を前記現在状態保持手段に追加して保持させることを特徴とする情報処理装置。

【請求項 34】 請求項 33 に記載の情報処理装置において、

前記状態合成手段の集合演算は、前記現在状態保持手段が保持する情報単位集合の選択情報である仕事名に関し、前記情報単位集合保持履歴記憶手段からその仕事名と同じ仕事名を有する他の利用者が生成した情報単位集合の内で利用者毎に最新の情報単位集合を抽出し、当該同じ仕事名を有する情報単位集合中の情報単位の和集合をとることを特徴とする情報処理装置。

【請求項 35】 請求項 33 に記載の情報処理装置において、

前記状態合成手段の集合演算は、前記現在状態保持手段の保持する情報単位集合の選択情報である 1 つの仕事名に関して、同じ仕事名を有する他の利用者が生成した情報単位集合の内の同じ仕事名を有する現在状態保持手段

の保持する情報単位集合よりも新しく、且つその利用者の情報単位集合の中で最新の第1情報単位集合を抽出し、さらに、その利用者のその仕事名を有する情報単位集合の中で同じ仕事名を有する現在状態保持手段の保持する情報単位集合よりも以前に作られた情報単位集合の中で最新の第2情報単位集合を抽出し、前記第1情報単位集合が保持する情報単位から前記第2情報単位集合が保持する情報単位の差分をとって第3情報単位集合を抽出し、さらに、当該第3情報単位集合が保持する情報単位から前記第1情報単位集合が保持する情報単位の差分をとることによって第4情報単位集合を抽出し、当該第4情報単位集合を前記現在状態保持部に追加して保持させることを特徴とする情報処理装置。

【請求項36】 請求項33に記載の情報処理装置において、前記状態成手段の集合演算は、前記現在状態保持手段の保持する情報単位集合の選択情報である1つの仕事名に関して、同じ仕事名を有する他の利用者が生成した情報単位集合の内の同じ仕事名を有する現在状態保持手段の保持する情報単位集合よりも新しく、且つその利用者の情報単位集合の中で最新の第1情報単位集合を抽出し、さらに、その利用者のその仕事名を有する情報単位集合の中で同じ仕事名を有する現在状態保持手段の保持する情報単位集合よりも以前に作られた情報単位集合の中で最新の第2情報単位集合を抽出し、複数の他の利用者のそれぞれについて前記第1情報単位集合が保持する情報単位から前記第2情報単位集合が保持する情報単位の差分をとって複数の他の利用者のそれぞれの第3情報単位集合を抽出し、さらに、それらの全ての第3情報単位集合の和集合をとって第5情報単位集合を抽出し、さらに、当該第5情報単位集合が保持する情報単位から前記第1情報単位集合が保持する情報単位の差分をとることによって第4情報単位集合を抽出し、当該第4情報単位集合を前記現在状態保持部に追加して保持させることを特徴とする情報処理装置。

【請求項37】 請求項33に記載の情報処理装置において、前記状態成手段の集合演算は、前記現在状態保持手段が保持する情報単位集合の選択情報に指定されている仕事名と同じ仕事名を有する他の利用者の生成した情報単位集合の内で利用者毎に最新の情報単位集合を抽出し、同じ仕事名を有する情報単位集合が保持する情報単位の共通集合をとることを特徴とする情報処理装置。

#### 【発明の詳細な説明】

##### 【0001】

【発明の属する技術分野】本発明は、電子的にアクセス可能な文書等のオブジェクト（情報単位）を保持する作業環境（情報単位集合）を提供することによって業務の遂行を支援する情報処理装置に関し、特に、複数の仕事を並行して行うために仕事との関連によりオブジェクト

を保持する作業環境を提供する情報処理装置に関する。

##### 【0002】

【従来の技術】今日では、パーソナルコンピュータやワークステーションなどの情報処理機器でファイル管理を行うことによって、様々なオフィス業務を遂行できるようになった。これらの機器が扱う文書、電子メール、アプリケーションプログラム、入出力装置などのリソースは、ファイルシステムやデータベース等によって管理される。このような管理において業務ごとに必要な文書一式と必要な道具をまとめておくことは、業務を円滑に進める上で効果的である。情報処理機器によって扱うことのできるリソースをまとめておくための従来技術には、階層型ディレクトリと、デスクトップによるものが知られている。

【0003】階層型ディレクトリは、例えば、UNIXやMS-DOS（登録商標）のようなオペレーティングシステムのファイルシステムによって採用されている。利用者は、情報をディレクトリという単位にまとめて保持することができ、さらに、複数のディレクトリを階層構造に構成することができる。階層型ディレクトリを採用した情報処理機器においては、利用者はディレクトリを業務に対応させて作成し、そこに関連するリソースを格納することができる。しかしながら、このような従来のファイルシステムでは、ディレクトリに格納された個々のリソースは、ファイル名のリストによって利用者に示されるに過ぎない。このため、どのような種類のリソースが業務で用いられるかどうかを直観的に理解するのが困難であった。

【0004】そこで、ゼロックス社のStar（商品名）によって、デスクトップと呼ばれる情報管理方法が開発された。デスクトップは、個人の作業環境を保持するものであり、メタファーとする机上を表現する二次元平面を提供する。業務に関連するリソースは、それぞれの種類に対応するアイコンと呼ばれる図形として二次元平面上に配置される。文書などのデータ、複数の文書をまとめるフォルダ、プリンタなどの周辺機器、アプリケーションプログラム、リファレンスなどが、利用者が直観的に理解しやすいシンボルとして配置される。なお、この他に、時計や計算機などの道具やその他のリソースも二次元平面上に配置される。これらのアイコンは、利用者によって自分の作業に適した位置に自由に配置することができる。更に、Starにおいては、ファイルサーバと呼ばれる分散ファイルシステムによって管理されているリソースを参照するために、リファレンスと呼ばれるアイコンを作成して、二次元平面上に配置することができる。リファレンスの機能によって、共有のリソースに対して個人の必要に応じて別名を与えることもできる。なお、当初、デスクトップは個人の作業環境として個人に対して1つ設けられているものであったが、リソースを配置するための二次元平面を複数用意する装置も

開発されている。

【0005】このような従来技術には、アップル社のMacintosh（登録商標）によるものや、ゼロックス社のRooms（商品名）によるものが知られている。Macintosh（登録商標）では、階層型ディレクトリによるファイルシステムにこの技術を適用している。各ディレクトリに対応したフォルダと呼ばれる二次元平面が管理され、ディレクトリが保持しているリソースをその上に配置することができる。そして、フォルダを開くと、さらに二次元平面の表示が得られる。ここで、タスクの切り替えに従って利用者に提示される情報を制御することを特徴とする作業域として、ワークスペースという概念がある。Bannonらは、"Evaluation and analysis of users' activity organization" (CHI'83)において、「オフィスにおける仕事は、並行して複数の業務を順繰りに行っており、前の仕事に戻るときは、以前それを行っていた環境が保存されているといい」といった、最初のワークスペースの概念を提唱した。

【0006】このワークスペースの概念を実現したものには、例えばゼロックス社のRooms（商品名）がある（特公平7-86820号公報）。Roomsでは、業務ごとにルームと呼ばれるデスクトップを定義することができ、ルームはリソースをまとめておくだけでなく、アプリケーションの実行状態も管理する。利用者は、それぞれのルームにおいて作業中のアプリケーションの状態を保存したまま、他の業務のためのルームに移動することができる。また、ドアとオーバービューという2つのメタファを用いて集合間に関係づけを行ったり、ルームの移動を行ったりすることもできる。

【0007】複数の業務作業のために複数の環境を用いる場合や、他の人に業務を引き継ぐような場合には、利用者がその環境で行なう仕事の構造や性質を想起或いは理解することが必要になる。前記した階層型ディレクトリやデスクトップにおいては、リソースの名前や種類などのリソース固有の情報、二次元平面での配置、他のディレクトリやデスクトップへの参照による階層構造を工夫することによって、利用者に仕事の構造を理解させる手助けとすることができる。例えば、リソースに適切な名前を与えたり、密接に関連するリソースを近くに配置することができる。

【0008】しかしながら、特定の業務においてリソースをどのように扱うか或いはリソース間にどのような関係があるかなどは、リソースの配置と名前或いはディレクトリやフォルダの階層構造だけで表現できるほど単純なものではない。Roomsでは、業務毎にルームを作り、各ルームには必要なリソースを配置することができ、そこで各業務を行うことができる。しかし、予め仕事に合わせて情報を分類しておくことは困難であり、また仕事の進行に応じて分類を組み替えるといったことは

非常に手間がかかる。そのため、どうしてもルームの設定がおおざっぱな分類になりやすく、仕事の体系とそぐわないという欠点がある。

【0009】業務における情報の構造を利用者にわかりやすく提示することのできる従来技術としては、例えば、特開昭60-108975号公報に記載されている発明がある。この発明では、利用者によって下絵画像の所定の領域が指示されると、その領域に対応づけられている別の下絵画像や文書等の計算機リソースを利用者に提示する。この発明ではワークスペースをガイドイメージと要素イメージの2階層で表す。要素イメージは仕事を行なうためのワークスペースの1つの状態を保持し、ガイドイメージは要素イメージの意味情報を持ち、要素イメージの切り替えを行なうためのガイドとなる。したがって、ガイドイメージ上に示されるタグの1つを選択することにより、要素イメージの表示を切り替えることができる。

【0010】また、ガイドイメージの表示には2つのモードがあり、共通イメージと個人イメージが選択できる。個人イメージを用いることにより、共通の情報を個人の観点で扱う枠組みを用意することができる。しかしながら、ガイドイメージはあらかじめ設定しておく必要があり、仕事を進めながら情報の分類体系を構築していくことは困難である。そのため、あらかじめ設定した業務の構造から外れずに一貫した分類を使用する場合はよいが、ダイナミックに変化する業務の体系に追従することは難しい。

【0011】また、「U.S. Pat. No. 5,159,669, AUTOMATICALLY CREATING A SECOND WORKSPACE OPERATION RECORD INCLUDING HISTORY DATA AND A UNIT ID BASED ON A FIRST WORKSPACE OPERATION", Trigg et al.」には、ワークスペースの利用履歴を再利用して新たなワークスペースを構築するというアプローチをとる発明が提案されている。この発明では、複数のカードへのリンクを保持できるカードをワークスペースとしてみなし、或るワークスペース上でどんなカードが作られたか、どんな変更が行われたか、といった一連の操作履歴のリストを表示するヒストリーワークスペースとして生成する。そのリスト中のカード名を表す部分がアンカーになり、そのヒストリーワークスペースの保持する操作履歴リストに現れるカードには容易にアクセスすることができる。つまり、一連の操作に関連する全てのワークスペース（カード）がヒストリーワークスペースとしてまとめて保存されることになる。しかしながら、この発明では操作履歴を保存することは可能になるが、残されたヒストリーワークスペースはおおよそ仕事を行う場と言えるのではなく、以前に行われた仕事と関連する情報を検索するため或いは関連するワークスペースを検索するためのアンカー集合である。

【0012】業務の進行状態の保存を行う従来技術には

以下のものがあるが、これらは過去のある場面を保存し、その場面からプログラミングなどを再開しようというものであり、仕事の環境を保持・再利用する機能はない。Windows 3.1（登録商標）には、スリープ機能としてデスクトップのダンプをとる機能がある。しかしながら、これは現在の環境を保存しておくことが目的で、複数の仕事の状態を保持することはできない。Smalltalk-80には、利用者の指示にしたがってスナップショットとして実行環境全体を保存することができる機能がある。しかしながら、この機能の狙いは仕事の特定の状態をスナップショットにより保存することであり、その状態を他の仕事で用いることにはない。したがって、複数の仕事は別のプロジェクトで行うよう設計されており、この観点では本質的にRoomsと同じである（Adele Goldberg原著、相磯秀夫監訳、"SMALLTALK-80 対話型プログラミング環境"、オーム社、1986.）。デバッグは、利用者の指示或いは設定にしたがってプログラムの実行環境を保存することができる機能である。しかしながら、この機能はプログラミングにおいて複数のモジュールの進行状態を保存するためのものであり、オブジェクトの保持状態を記憶する本発明とは目的・構成ともに異なる。

【0013】履歴情報を再利用する従来技術には以下のものがあるが、これらは履歴を利用してオペレーションを再利用する、履歴を用いて情報を検索する、といった履歴を単なる過去の状態としか捉えていない。そのため、過去の或る時点での状態を復元し、そこから新たな仕事を始めるということができない。特開平6-266662号公報に記載される協調作業支援装置では、協調作業の履歴から過去の作業内容を正確に再現でき、その後の協調作業の能率向上に役立てることができる。しかしながら、狙いとする機能は作業内容を再現するのみで、過去の仕事の状態をこれからの仕事に活かすことはできない。特開平6-139117号公報に記載されるヒストリ装置では、操作履歴のヒストリとともに操作時の画面イメージを表示することによって、ヒストリの選択を容易にしている。すなわち、操作の履歴を記録する際に、そのとき表示されている画面のビットマップイメージを取り込んでおき、ヒストリを表示する際に、画面イメージを同時に表示する。しかし、これは過去の操作の意味を思い出しやすくするだけで、その状態から仕事を再開することはできない。

【0014】特開平6-214989号公報に記載される文書表示方法及びその装置では、ハイパーテキスト文書において、利用者が参照したノードやその参照経路についての経緯を容易に把握させている。すなわち、ノード間移動履歴を蓄積しておいて、文書変更或いは参照された順にノードとリンクとを表す記号を表示し、この上で履歴マークを履歴に合わせて時系列順に移動表示することにより、経緯を示している。しかし、これは履歴を

時系列に表示するだけで、その状態から仕事を再開することはできない。

【0015】

【発明が解決しようとする課題】情報を組織的に共有・活用していくためには、お互いの情報を正確に理解することが重要である。そのためには、組織内で保有されている情報が、どんな仕事のどんな状況で作られたものかという背景情報を知ることが不可分である。情報が作られた背景情報を知るためには、業務の視点で情報が整理されている必要がある。しかしながら、業務が進行するにつれて情報の分類や体系は動的に変化していくものであり、これに追従して業務を進めながら柔軟に情報を分類・整理していくことは困難である。このような状況から、下記の（1）及び（2）に挙げる課題があった。

【0016】（1）業務に合わせた情報の分類・整理が困難：業務ごとに関連する情報を管理するためには、情報に対して各業務に応じた観点で意味付けを行うことができない。そのため、仕事の体系に合わせて情報を分類・整理し、体系的に管理する必要がある。また、複数の業務を並行して行うことを支援するためには、業務毎に整理された情報を提供する環境を複数持ち、それぞれの業務毎に進行状態を保持し、保持された業務を利用者が自由に選択できる必要がある。また、これまで行なっていた業務の分類を離れ、新たな観点から業務を開始しようとした場合には、分類のやり直しには多大な労力が必要であった。

【0017】（2）業務を通じて得られるノウハウの再利用が困難：或る仕事にはこの文書を使う、或る仕事はこの人に結果を送る、或る仕事はこういった処理をするなど、仕事には多くのノウハウがある。それらノウハウのうちで定型化できるものは、マニュアル化して共有を行なっている。しかしながら、業務を通じて得られる多くのノウハウは定型化することが困難で、そのため共有することが困難である。このため、このような仕事の実行過程を通して得られるノウハウを再活用するための枠組が必要であった。

【0018】本発明は上記従来の事情に鑑みなされたもので、作業環境において行なわれた活動を反映し、活動に必要な情報を保持する作業環境を自動的に構築する情報処理装置を提供することを目的とする。また、本発明は、上記の作業環境を構築するとともに、業務毎に情報を整理する情報処理装置を提供することを目的とする。また、本発明は、上記の作業環境を構築するとともに、業務毎の進行状況を保存する情報処理装置を提供することを目的とする。また、本発明は、仕事に応じて過去の状態を復元或いは過去の状態から必要な情報を取り出して、上記の作業環境を構築する情報処理装置を提供することを目的とする。また、本発明は、構築された上記の作業環境を用いて新たな仕事の開始を可能ならしめる情報処理装置を提供することを目的とする。



## 【0019】

【課題を解決するための手段】本発明に係る情報処理装置では、上記の作業環境を情報単位集合とし、この作業環境で利用されるデータオブジェクトやアプリケーションオブジェクト等を情報単位とし、情報単位集合保持履歴記憶手段に複数の情報単位集合を記憶する一方、情報単位集合保持手段に作業対象となる情報単位集合を保持させる。そして、仕事の変更、利用者の変更、情報単位の追加や削除等といった所定の条件が整うと、記憶指示手段により情報単位集合保持手段が保持する情報単位集合を情報単位集合保持履歴記憶手段に記憶せしめるとともに、この情報単位集合保持履歴記憶手段へ記憶される情報単位集合に、選択情報付与手段が仕事名、利用者名、オブジェクト名、日時等といった選択情報を付与する。また、或る情報単位集合を作業対象とするために、利用者から選択情報が入力された場合等においては、選択手段が選択情報付与手段により付与された選択情報を用いて、情報単位集合保持履歴記憶手段に記憶された情報単位集合の中から該当する情報単位集合を選択し、この情報単位集合が読出手段によって情報単位集合保持履歴記憶手段から情報単位集合保持手段へ読み出される。

【0020】したがって、本発明の情報処理装置では、利用者がその時点で必要な情報をワークスペースに集めるという作業により、前もって分類を行なわなくとも業務の進行状況に応じて動的に情報単位集合（作業環境）を蓄積管理する。そして、このような過去の仕事の状態に応じた情報単位集合を選択情報を付随させて保持しているため、選択情報に基づいて読出を行うことにより、過去の時点の状態に戻って仕事をスタートさせることや、過去の状態を復元して仕事に必要な情報を抜き出すことができる。

【0021】また、本発明に係る情報処理装置では、情報単位集合中に含まれる情報単位の数が変更された場合に、変更前の情報単位集合を保存するようにしており、これによって、変更前の状態で情報単位集合を何時でも復元することを可能ならしめている。例えば、情報単位集合中に含まれる情報単位の数が増加から減少に転じた時点仕事を区切りとして犠牲し、この仕事の区切り前の状況を何時でも復元することを可能ならしめている。更に、本発明に係る情報処理装置では、このような機能をより充実させるために、下記のような機能を備えている。

【0022】（1）現在の仕事の場の状態を保持するだけでなく、過去の仕事の場の状態の遷移を時系列に記憶する。

（2）仕事の場における情報単位の増減など、仕事の場における様々なオペレーションを監視し、あらかじめルールに定義されたタイミングで仕事の場の状態の記憶を行う。

（3）情報単位集合の記憶時に付与される選択情報を利用

いて、状態遷移を利用者に提示する。

（4）利用者が提示された状態遷移から選択情報を指示すると、その条件を満たす最新の仕事の場の状態（情報単位集合）を読み出し、現在の仕事の場の状態にする。

（5）仕事の場の状態を指示し、その状態が保持する情報単位から一定の条件を満たす情報を抽出する。

（6）複数の仕事の場の状態を指示し、それらの保持する情報単位の和集合、共通集合、差分等を演算して、新たな仕事の場を作成する。

【0023】本発明に係る情報処理装置では、利用者が複数の業務を並行して行う際に、業務に必要な情報を集めるという自然なアクションの結果を反映させて、業務を進めるに従って次第に情報の分類整理がなされる。したがって、整理の上手でない人でも、仕事を進めながら順次必要なオブジェクトを集めてグループ化していくことが可能になり、これによって情報の分類がうまくいくようになる。Roomsとの比較で述べるならば、Roomsでは作業環境を仕事を行う前に作成する必要があるが、本発明では仕事が進んだ後でも作業環境を視点を変えて切り直すことができる。業務体系は仕事を進めながら変化していくものだが、本発明では、仕事を進めながら「仕事の切り替え」と考えられるタイミングで装置が必要な選択情報とともに仕事の場の状態を記憶する。利用者は、今進めている仕事名を入力したり、他の仕事を始めるときにその仕事名を選択したり、といった形で仕事を行い、この仕事の進行に伴って、同じ仕事の場にあった情報単位は次第にグループ化されて仕事に合わせて整理されることになる。

【0024】また、仕事の場の状態の記憶は利用者が仕事の切り替え時と認識した時点のみに行われるのではなく、例えば仕事の場における情報単位の増減のタイミングで自動的に行われ、仕事を進めてしまった後で過去の仕事の遷移から分岐して新たな仕事を始めることが可能である。すなわち、作業を行った後、それまでの作業で無意識的に蓄積された情報に対する意味付け（例えば、或る情報はどの仕事にも使われるとか、或る情報は或る仕事に特有のものであるとか、或る関連の仕事に必要な情報は何かといったこと）等のノウハウを抽出し、後の作業に再利用することができる。また、本発明では、仕事の場の「或る一場面」を仕事の切り替えである可能性のあるタイミングで随時記憶し、その時に表明されている仕事の種類、利用者名、仕事の切り替えのトリガとみなされる流入リソース等といった属性を付与して記憶しておく。そのため、仕事の種類で場面を構造化すれば最低限Roomsと同じ機能を提供することができる。

【0025】また、本発明では、仕事の場の各場面を「情報単位の集合」として保存しているため、情報単位集合の検索を行ったり、集合演算による新たな情報単位集合の生成を行うことができる。例えば、2つの原稿を別々の作業環境で書いており、後からそれらを1つの原



稿にしようと思った場合には、和集合をとることにより両方の原稿が保持する情報がマージされて新たな原稿が作られる。また、いくつかの作業環境で異なる原稿を書いていた場合に、それらの作業環境の共通集合をとることにより、原稿を書くために必要な基本環境を抽出することができる。また、或るテーマの原稿を書いている作業環境に情報が溢れてきてしまった場合、当該テーマについての報告書を書いている他の作業環境があったとすると、それらの作業環境の差分をとることにより、当該テーマの原稿を書くための作業環境に特有の情報が何かなを知ることができる。

#### 【0026】

【発明の実施の形態】本発明に係る情報処理装置の実施例を図面を参照して説明する。図1には、本発明の第1実施例に係る情報処理装置の構成を示してある。この情報処理装置は、ネットワーク10を介して接続されたサーバ1とクライアント11により構成されており、サーバ1及びクライアント11はどちらもワークステーション或いはパーソナルコンピュータにより構成されている。なお、本実施例ではクライアント・サーバによる実現形態をとっているが、このような構成である必要はなく、例えばネットワークを用いずに1台の計算機内に情報処理装置を構成してもよい。

【0027】サーバ1は、情報単位記憶部2、情報単位集合保持履歴記憶部3、情報単位集合選択部4、情報単位集合読出部5、選択情報付与部6、記憶指示部7を有しており、記憶指示部7は情報単位集合記憶判断部8と操作コマンド蓄積部9とを有している。なお、サーバ1は複数の利用者（クライアント）によって共通に用いられる情報を管理する。情報単位記憶部2は、過去に作られた情報単位を書き込み読み出し自在に記憶する。ここに、情報単位は、参照オブジェクトに対して仕事における意味付けを与えるためのもので、参照オブジェクトへのポインタと、参照オブジェクトに対するコメントなどの付加属性を有している。

【0028】情報単位集合保持履歴記憶部3は、過去に記憶された複数の情報単位集合を選択情報とともに書き込み読み出し自在に保持し、本実施例では、関係データベースにより情報単位記憶部2と情報単位集合保持履歴記憶部3とが構成されている。ここに、情報単位集合は文書やアプリケーションなどを参照するための情報単位の集合を保持し、例えば2次元空間上（デスクトップメタファ）或いはリストとして情報単位を複数配置できる作業空間である。選択情報付与部6は、情報単位集合保持履歴記憶部3に記憶された情報単位集合に対して、ジョブ名或いは日時・作成者名・情報単位名などの選択情報を付与する。

【0029】情報単位集合選択部4は、情報単位集合保持履歴記憶部3に記憶されている情報単位集合の1つを選択情報に基づいて選び出す。情報単位集合読出部5

は、情報単位集合保持履歴記憶部3に記憶された1つの情報単位集合をクライアント11の情報単位集合保持部14へ読み出す。記憶指示部7は、情報単位集合記憶判断部8と操作コマンド蓄積部9から成り、情報単位集合を情報単位集合保持履歴記憶部3に記憶する指示を与える。

【0030】情報単位集合記憶判断部8は情報単位集合の記憶を行なうかどうかの判断を行い、操作コマンド蓄積部9は利用者による操作の指示内容を蓄積する。すなわち、情報単位集合記憶判断部8は、利用者による操作を監視し、情報単位の作成・削除、情報単位集合の切り替えといった指示が行われる度に操作コマンドを操作コマンド蓄積部9に蓄積し、さらに、あらかじめ情報単位集合の記憶のタイミングを設定した記憶実行ルールと照らし合わせるにより、情報単位集合の記憶を行なうかどうかの判断を行ない、情報単位集合保持履歴記憶部3へ記憶実行の指示を行なう。例えば、情報単位集合が保持する情報単位が増減したタイミングなどで記憶実行の指示が行われる。

【0031】クライアント11は、入出力部12、情報単位集合表示部13、情報単位集合保持部14、情報単位編集部15、選択情報リスト表示部16、選択情報指示部17を有している。なお、クライアント11は利用者が仕事を行うためのもので、ワークステーションやパーソナルコンピュータ上のソフトウェアによって実現されている。入出力部12は、ディスプレイ、キーボード、マウスなどの入出力装置によって構成される。ここでは、各種のアプリケーションに共通に用いられるウィンドウの表示、キーボード入力、カーソルによるアイコンやボタン操作などの標準的な入出力制御が行なわれるものとする。

【0032】情報単位集合表示部13は、情報単位集合保持部14に保持された情報単位集合をディスプレイ画面に表示出力する等して利用者に提示する。情報単位集合保持部14は、現在利用者が作業を行なっている環境である情報単位集合を書き込み読み出し自在に保持する。情報単位編集部15は、入出力部12から入力された利用者の指示に従って、情報単位集合保持部14が保持する情報単位の配置を変更したり、情報単位を情報単位集合保持履歴記憶部3が記憶する他の情報単位集合にコピーしたり、情報単位集合保持部14が保持する情報単位集合を情報単位集合保持履歴記憶部3へ格納させたり、情報単位にコメントを加えたりする。

【0033】選択情報リスト表示部16は、情報単位集合保持履歴記憶部3に記憶されている情報単位集合に付与されているジョブ名、或いは日時・作成者名、情報単位名などの選択情報をディスプレイ画面に表示出力する。なお、利用者はリスト表示される選択情報のフォーマットを入出力部12から指示を入力して選ぶことができる。選択情報指示部17は、選択情報リスト表示部1

6に表示された選択情報から、情報単位保持部14に保持する情報単位集合として読み出すための条件を指示する。なお、利用者は入出力部12を通して選択情報の条件を指示することができる。

【0034】すなわち、上記構成の情報処理装置では、利用者が選択情報指示部17によって、クライアント11の選択情報リスト表示部16に表示された選択情報を指示すると、これに対して、サーバ1が情報単位集合保持履歴記憶部3に記憶された情報単位集合のうちで指定された選択情報をすべて満たす情報単位集合を検索する。そして、選択情報をすべて満たす情報単位集合の中で、最も作成日時が新しいものについて、読出部5がクライアントの情報単位集合保持部14に読み出して保持させる。そして、情報単位集合表示部13が情報単位集合保持部14に保持された情報単位集合を表示出力する。

【0035】ここで、本発明の動作原理を説明するために必要な「ワークスペース」、「情報単位」、「情報単位集合」、「ジョブ」の概念を定義しておく。「ワークスペース」は、利用者にとっての仕事の場を表す概念である。ワークスペースは、利用者の現在の仕事の場の状態を保持するカレント情報単位集合と、その仕事の場の過去の状態を保持する情報単位集合を含む。ワークスペースでの仕事は、カレント情報単位集合を更新していくことで遂行され、過去の情報単位集合をカレント情報単位集合にすることで、過去の仕事を継続して遂行することができる。利用者は、仕事の種類ごとにワークスペースを複数作成してもよいし、全ての仕事を行なうための1つのワークスペースを作成してもよい。

【0036】「情報単位」は、仕事に必要な情報・文書・アプリケーションなどのオブジェクトを参照し、仕事における意味付けの情報と対応づける単位である。情報単位の仕事におけるオブジェクトの意味付けのための情報には、オブジェクトへのポインタ、利用者の付与したコメント、ワークスペース上の配置などがある。「情報単位集合」は、ワークスペースのある時点の状態を表し、情報単位の集合である。情報単位集合が保持する情報単位の集合は、仕事の進行とともに増減する。つまり、情報単位集合は、ある時点での利用者に提示された作業環境を示す。ワークスペースは、仕事の履歴として情報単位集合を複数記憶し、カレント情報単位として復元できるように管理し、情報単位集合の記憶は、例えば情報単位集合の保持する情報単位が増減したとき、といった仕事の状態変化をトリガに行われる。利用者により選択情報が指示されると、記憶された情報単位集合から選択条件にマッチするものを1つ検索し、それをカレント情報単位集合とする。利用者は読み出された環境で仕事を継続することができる。

【0037】「ジョブ」は、ワークスペースにおいて行われる仕事の論理的なまとまりを表す概念である。ジョ

ブは、同じ仕事に属する全ての情報単位集合の集合である。利用者がジョブ名を宣言すると、それ以降記憶される情報単位集合とジョブが関連付けられる。逆に選択条件としてジョブを指定すると、そのジョブに属する最新の情報単位集合をカレント情報単位集合として読み出すことが可能になる。つまりジョブ名は、仕事の切り替えのインデックスとして機能する。

【0038】図2には、本実施例におけるワークスペースを用いた仕事の場の状態の遷移の管理モデルの概要を示してある。情報単位集合保持部14は、現在の利用者に表示されているワークスペースの状態、つまりカレントの情報単位集合を保持する。例えばクライアントのメモリ上などに保持される。情報単位集合保持履歴記憶部3は、利用者の仕事の進行に伴って変化するワークスペースの任意の状態、つまり過去の情報単位集合を復元可能にするため、随時必要な情報単位集合を記憶する。記憶指示部7は、情報単位集合保持部14の保持する情報単位集合における情報単位の増減などといったトリガにより、情報単位集合の情報単位集合保持履歴記憶部3への記憶指示を行なう。利用者によってジョブ名・情報単位名・利用者名・日時といった選択情報が指示されると、情報単位集合選択部4は情報単位集合保持履歴記憶部3に記憶されている情報単位集合を選択し、それを読出部5により復元、すなわちカレント情報単位集合として情報単位集合保持部14に読み出す。

【0039】このように、ワークスペースのある時点の状態（情報単位集合）に選択情報を付与する機能、その状態を呼び出して復元し、仕事を再開する環境を提供する機能を有している。この機能により、利用者はワークスペースの状態を切り替えながら複数の仕事を遂行することができる。また、本発明では、利用者は、ある仕事に必要な情報を特定のジョブ名の情報単位集合に集めることによって、その仕事を遂行することができる。つまり、利用者は、仕事を始める前にあらかじめ情報の分類を考慮しなくても、仕事に必要な情報単位の集合に対してジョブ名を付けていくことにより、次第に情報単位を仕事の観点で整理していくことができる。

【0040】また、利用者はワークスペース上での作業中に任意のタイミングでジョブを宣言できる。ジョブの宣言はメニューやボタンを利用者が操作することによって行われ、これによりジョブ名の入力が促され、利用者によりジョブ名が入力されることで、新しいジョブが生成される。ジョブが宣言されると、情報単位集合保持履歴記憶手段3は、カレント情報単位集合の選択情報としてそのジョブ名を付与する。

【0041】また、選択情報指示部17により利用者がジョブを指定すると、システムは所望のジョブ名を持つ最新の情報単位集合を呼び出す。そして、ワークスペースのカレントの情報単位集合が、呼び出された情報単位集合に書き換えられ、それが表示されることにより、利

用者はジョブに対応する仕事を再開することができる。これにより、利用者はワークスペース上に表示される情報単位集合を切り替えながら、同一の作業環境下で並行して複数の仕事を行うことができる。つまり、Roomsのように仕事毎にルームをあらかじめ定義しておかなくても、各々の仕事を行っている時に情報単位集合にジョブ名を付け、これを切り替えることで仕事に対応する作業環境への移動の効果をすることができる。

【0042】図3には、情報単位集合が情報単位集合保持履歴記憶部3に記憶され或いは読み出しが行われる際のデータと処理の流れの概要を示してある。すなわち、選択情報リスト表示部16に表示された選択情報の内の1つを利用者が指示すると、カレントの情報単位集合にそれまでのジョブ名などの選択情報を付与し、カレントの情報単位集合を情報単位集合保持履歴記憶部3へ記憶実行する。続いて、指示された選択情報により選ばれた情報単位集合がワークスペース上に取り出され、その情報単位集合がワークスペースのカレント情報単位集合になり情報単位集合表示部に表示され、そこから仕事を再開することができる。

【0043】図4には、ワークスペースでの操作と、それによって格納される情報単位集合との関係の例を時系列に沿って示してある。まず、ワークスペースが新規作成されると、そこではまだ何の仕事を行っているかという表明がなされていないので、新たな情報単位が流入した（つまり仕事を始めた）直後に、その仕事の種類（ジョブ名）の入力を利用者に促す。そして、“job1”というジョブ名が入力されたとすると、次から記憶される情報単位集合には、“job1”という選択情報が付与される。その後、新しい情報単位が流入したときには、そのときの情報単位集合を記憶する。図4では、情報単位集合保持履歴記憶部3に記憶された情報単位集合は“set1”と表されている。

【0044】その後、次のジョブ名が宣言される、或いは他のジョブが選ばれるといったことが行われるまでは、その後の作業に伴って記憶される情報単位集合は全て“job1”というジョブ名が付けられる。この例では、新たなジョブが宣言されるまでに4つの情報単位集合（“set2”、“set3”、“set4”、“set5”）の記憶が行われ、ジョブ名“job1”の情報単位集合が計5つ記憶されている。そして、利用者がそれまで行っていた“job1”という仕事を中断し、次に“job2”という仕事を始めた（ジョブ名“job2”が表明された）時点で、その時の状態を記憶し、さらに、その後に記憶される情報単位集合のジョブ名を変える。

【0045】なお、再び“job1”の仕事の再開が指示された場合には、ジョブ名“job1”の最新の情報単位集合が検索され、その情報単位集合がワークスペースのカレント情報単位集合となり、情報単位集合表示部

13に表示され、利用者は仕事の再開が可能になる。もちろん、この後の情報単位集合の記憶にあたっては、“job1”というジョブ名が付けられる。

【0046】図5には、ディスプレイ画面上における表示の一例を示してある。この表示例では、情報単位を配置する情報単位集合表示領域21と、ジョブを表示・選択する選択情報リスト表示領域22とが形成されている。情報単位集合表示領域21は、“Printer”、“Doc-4”、“特許DB”等といった情報単位を2次元平面上に配置する。利用者は、各情報単位をマウスなどのポインティングデバイスにより指示することにより、参照するオブジェクトを立ち上げたり、付加属性の編集を行ったり、配置を変更したりすることができる。

【0047】選択情報リスト表示領域22は、情報単位集合を選択するためのジョブ名や利用者名などの選択情報を表示する領域である。選択情報としては、宣言されたジョブ、情報単位集合の記憶を行なったときの利用者、情報単位集合の記憶を行なったときの日時、情報単位集合の記憶を行なう直前に生成されたオブジェクト、といったものがある。図5の例では、選択情報として“Database特許”、“情報検索特許”といったジョブ名が表示されるように設定されている。現在の情報単位集合表示領域21に表示されている情報単位集合がどのような仕事に関するものかを知るために、利用者により指示された選択情報は選択情報リスト表示領域22内で反転表示される。選択情報リスト表示領域22のジョブ名を選択することにより、情報単位集合表示領域21に提示される情報単位集合を切り替えることができる。これにより、ジョブを指示して情報単位集合を切り替えながら、並行して複数の仕事を進めることが可能になる。

【0048】また、情報単位集合にはジョブ名の他に、記憶されたときの利用者名や、流入した情報単位名などの選択情報が付与される。例えば、図6にはジョブ名と利用者名で情報単位集合を構造化して表示した例を示してある。ジョブ“Database特許”は利用者“Aoki”と“Suzuki”の2人で仕事を進めていることが解り、また、ジョブ“情報検索特許”は“Yamada”も加わって仕事をしていることが解る。図6のように、このリストの“Database特許”の“Aoki”を指定することにより、ジョブ名“Database特許”で作成者“Aoki”の最新の情報単位集合が呼び出される。これは、Aokiさんが“Database特許”という仕事を行っている最新の状況と呼び出すことであり、つまりAokiさんは常に自分の名前を指定することにより、Suzukiさんの作業の影響を受けずに仕事を行うことが可能になる。もちろん、Suzukiさんは“Suzuki”を指定すれば自分の行っている最新の状況と呼び出すことができるし、A

o k iさんの現在の進行状況を知ること容易である。また、利用者に関係なく、この仕事の最新の状況を知りたい場合は、ジョブ名"Database特許"を指定することにより、利用者と無関係に最新のDatabase特許の情報単位集合を呼び出すこともできる。

【0049】また、例えば図7に示すように、ジョブ名に加えて、流入した情報単位名を使って情報単位集合を構造化して示すことにより、より詳細な選択情報の指示が行えるようになる。例えば「最近1週間に流入した情報単位に限定して表示」といった制限をかけておけば、最近の仕事の進行状況を任意の場面の様子を参照しながら確認することができる。

【0050】次に、本実施例における具体的なデータ構造を示す。この例では、情報単位や、情報単位の参照するオブジェクトのバージョン管理については言及していないが、従来用いられているバージョン管理技術を用いることにより、情報単位やその参照先オブジェクトの変更を全て記録しておくことも考えられる。こうすることにより、呼び出された履歴中の情報単位集合に含まれる情報単位やその参照先オブジェクトの状態を過去のその状態にそのままに復元することができる。このような機能を実現することにより、共同で執筆活動をする際の支援を行うことができる。

【0051】図8には、利用者名(ユーザ名)を管理するためのデータ構造の一例を示してあり、利用者の管理はid、名前、パスワードによって従来の管理技術により達成される。図9には、ワークスペースのデータ構造の一例を示してある。ワークスペースは、id、名前、現在ワークスペース領域に表示されているカレント情報単位集合のid、選択情報リストの表示フォーマットのidを有している。ワークスペースは複数の情報単位集合を保持する主体であり、ワークスペースをどのように分けて持つかということについては、使い方はいくつかある。例えば、1人に1つのワークスペースという使い方もできるし、論理的なタスクに対してワークスペースをそれぞれ割り当てるという使い方もできる。

【0052】図10には、ジョブのデータ構造の一例を示してあり、ジョブはid、名前、作成日時、作成者名を有している。ジョブはワークスペース中で利用者が宣言した仕事の名前であり、利用者が宣言する度にジョブデータが生成される。図11には、ワークスペースとジョブの対応を表すデータ構造を示してあり、どのワークスペースがどのジョブを含むのかという対応をidによって表すテーブルである。なお、ジョブが生成される度に対応データが追加される。図12には、情報単位集合のデータ構造の一例を示してある。情報単位集合はid、所属するワークスペースのid、作成日時、作成者の他に、この情報単位集合のジョブ名、この情報単位集合に流入した情報単位idといった情報を有している。

【0053】図13には、情報単位のデータ構造の一例

を示してあり、id、名前、作成日時、作成者の他に、参照オブジェクト、起動アプリケーション、付与されたコメント、ワークスペース上の配置といった情報を有している。図14には、ワークスペースと情報単位集合の対応を表すデータ構造を示してあり、どのワークスペースがどの情報単位集合を含むかという対応をidによって表すテーブルである。なお、情報単位集合が生成される度に対応データが追加される。図15には、情報単位集合と情報単位の対応を表すデータ構造を示してあり、どの情報単位集合がどの情報単位を含むかという対応をidによって表すテーブルである。

【0054】また、選択情報リストの表示のために、図16に示す選択情報のテーブルと、図17に示す選択情報リストのテーブルを有している。図16の選択情報テーブルは選択情報としてジョブ、作成者、流入情報単位、作成日時が利用できることを示しており、図17の選択情報リストテーブルは選択情報リスト表示部16の表示フォーマットを保持している。例えば、図17のリストテーブルは、第1階層の選択情報として、選択情報idが1(つまり、図16からジョブ)が設定されていて、第2階層の選択情報として、選択情報idが3(つまり、図16から流入情報単位)が設定されていることを示す。

【0055】次に、情報単位集合を記憶する際のルールを説明する。情報単位集合の情報単位集合保持履歴記憶手段3への記憶は、利用者の指示を含む操作コマンド列が特定の組み合わせになったことを契機に実行される。記憶実行のタイミングは、記憶実行ルールの設定により様々なバリエーションで行なうことができる。利用者がワークスペース上でオペレーションを行なうと、まず、その操作コマンドが操作コマンド蓄積部9に蓄積される。次に、記憶実行ルールと蓄積されたコマンドとが参照され、情報単位集合を記憶実行するかどうかの判断が判断部8によって行われる。

【0056】具体的には、利用者が「情報単位集合記憶」という直接的な指示をボタンなどで行う、或いは、新規ジョブの表明、ワークスペース上の情報単位に対するオールクリア、情報単位集合への情報単位の流入といった仕事の切り替えに関わるタイミングを検知し、その操作コマンドを図18に示すように操作コマンド蓄積部9に蓄積する。操作コマンド蓄積部9は利用者の操作を順に、例えば、ワークスペースを開いた(OpenWS)、ワークスペースを閉じた(CloseWS)、ジョブを選んだ(SelectJob)、カレント情報単位集合を切り替えた(SwitchJob)、情報単位を削除した>DeleteUnit)、情報単位を作成した(CreateUnit)等の操作内容を蓄積する。

【0057】そして、情報単位集合を格納するかどうかを決定するためのルールは例えば図19に示すように設

定される。すなわち、ワークスペースで新たな仕事を始めた、或いは、仕事を切り替えたといったタイミングでは必ず直前の仕事の状態が保存される。仕事の切り替えのタイミングとして考えられるものには、新規作成したワークスペースの情報単位集合に情報単位を取り込んだとき、カレント情報単位集合を保持する情報単位を全て削除してから情報単位を生成したとき、ジョブを指定してカレント情報単位集合を切り替えた後で情報単位を生成したとき等がある。これらのタイミングは、新しい仕事が始まるタイミングと考えられ、このような仕事の変化とみなされるタイミングで情報単位集合の記憶が指示される。

【0058】次に、本実施例における処理動作を図20乃至図24に示すフローチャートを参照して説明する。図20には、情報単位を情報単位集合に取り込む情報単位の取り込み処理(A)の処理手順を示してある。情報単位の取り込みは、例えばOSが提供するファイル管理ツールからコマンド或いはポインティングデバイスによる指示によって行われる。また、他のアプリケーションの指示により取り込みが行われる、或いは、情報処理装置のコマンドによりファイル管理ツールで取り込みが行われるといった方法がある。情報単位の取り込みの指示があると、情報処理装置は、まず、その指示を行なった利用者の利用人名、その日時、流入するファイル名といった情報を取得する(ステップS1)。

【0059】次いで、情報単位のデータテーブルに新規データを生成し(ステップS2)、操作コマンド蓄積部9に"CreateUnit"を追加した後(ステップS3)、図21に示す記憶判断・実行処理(B)へ進む(ステップS4)。そして、記憶判断・実行処理(B)の後に、カレントの情報単位集合に新規生成した情報単位を追加する(ステップS5)。すなわち、情報単位集合-情報単位対応テーブルにカレントの情報単位集合のidと新規作成した情報単位のidの対応を加える。その後、この情報単位集合の再表示を行って処理を終了する(ステップS6)。

【0060】図21には、情報単位集合の記憶判断・実行処理(B)の処理手順を示してある。この処理は、蓄積された操作内容(コマンド)が記憶実行ルールに適合するかどうかの判断を行ない、適合していればルールに沿って記憶を実行する。例えば、利用者による情報単位の生成・削除・移動・編集、情報単位集合の記憶指示・読み出しなどのオペレーションの指示が行われたときに呼び出される。すなわち、図18に示したように蓄積された操作コマンド列の状態が、図19に示した記憶実行ルールを満たしているかどうかをチェックする。本実施例では、Rule-1から順に、ルールを満たしているかどうかをチェックし、適合するものがあれば記憶実行する。図19に示したRule-1は「新規作成されたワークスペースに、新たに情報単位が生成されたとき」

を表す。この場合、操作コマンド蓄積部9に蓄積されたコマンド列の末尾から2番目にワークスペースが新規作成された(CreateWS)、末尾に情報単位が作られた(CreateUnit)というコマンドが続いていれば条件を満たすので、情報単位集合の記憶実行が行われる。

【0061】すなわち、記憶実行ルールを満たしているかをチェックし(ステップS10)、記憶実行のルールを満たしていた場合には、ジョブ名が宣言されているかどうかをチェックし(ステップS11)、ワークスペースを作成したばかりでジョブ名が宣言されていなければ図22に示す新規ジョブ登録処理(C)へ進む(ステップS12)。そして、ジョブ名が宣言されている場合には、情報単位集合保持部14に保持されている情報単位集合を情報単位集合保持履歴記憶部3にコピーし、作成者・日時・ジョブ名・オブジェクト名の選択情報を付与する(ステップS13)。次いで、情報単位集合保持部14に保持されるカレント情報単位集合のidを新しいものに変えた後(ステップS14)、ワークスペース内に記憶されている情報単位集合を管理するためのワークスペース-情報単位集合対応テーブルに新規作成した情報単位集合のidを加え(ステップS15)、処理を終了する。

【0062】図22には、新規ジョブ登録処理(C)の処理手順を示してある。この処理は、記憶判断・実行処理(B)から、或いは、新規ジョブの宣言が利用者により表明されたときに呼び出される。情報処理装置は、まず、ジョブ名入力ウィンドウをディスプレイ画面に開いて利用者に対してジョブ名の入力を促す(ステップS20)。そして、ジョブ名が入力されると(ステップS21)、当該ジョブ名が同じワークスペース内で使われているかどうかチェックし(ステップS22)、使われていれば「そのジョブ名は既に使われております」とディスプレイ画面に表示してジョブ名の入力を促すステップへ戻る。

【0063】そして、新規のジョブ名が入力された場合には、新規のジョブデータを生成し(ステップS23)、操作コマンド蓄積部9に"CreateJob"を追加し(ステップS24)、記憶判断・実行処理(B)を呼び出す(ステップS25)。そして、情報単位集合保持手段14が保持するカレントの情報単位集合のジョブ属性を新規のジョブのidに変え(ステップS26)、ワークスペース内のジョブを管理するワークスペース-ジョブ対応テーブルに新規ジョブのidを加える(ステップS27)。この後、選択情報リスト表示部16による再表示を行なうため、図23に示す選択情報の分類表示処理(D)へ進む(ステップS28)。

【0064】図23には、選択情報の分類表示処理(D)の処理手順を示してある。この処理では、まず、図17に示した選択情報リストテーブルを参照し、第1

階層と第2階層の選択情報属性を獲得する(ステップS30)。そして、ワークスペース-情報単位集合対応テーブルを参照し、ワークスペースが保持する全ての情報単位集合のidを獲得し(ステップS31)、それらの情報単位集合を作成日時でソートする(ステップS32)。次いで、それらを基に、第1階層-第2階層の選択情報の属性値の対のリストを生成し(ステップS33)、カレントの情報単位集合に相当する選択情報を示すため、ワークスペースのカレント情報単位集合を参照し(ステップS34)、カレント情報単位集合に対応する第1階層-第2階層の選択情報属性を特定する(ステップS35)。この後、選択情報の属性値の対のリストを表示し(ステップS36)、カレント情報単位集合に対応する選択情報の属性値を反転表示する(ステップS37)。

【0065】図24には、仕事の場の状態の切り替えに際して行われる情報単位集合の読み出し処理(E)の処理手順を示してある。この処理は、選択情報リスト表示部16の選択情報が利用者により指示されると呼び出され、まず、指示された選択情報を獲得する(ステップS40)。そして、操作コマンド蓄積部9に"Select Job"を追加し(ステップS41)、記憶判断・実行処理(B)を呼び出す(ステップS42)。

【0066】そして、選択情報の条件を満たす最新の情報単位集合を検索し(ステップS43)、操作コマンド蓄積部9に"Switch Set"を追加し(ステップS44)、記憶判断・実行処理(B)を呼び出す(ステップS45)。次いで、検索された情報単位集合を情報単位集合保持部14に読み出し(ステップS46)、検索された情報単位集合のidをワークスペースが保持するカレント情報単位集合にする(ステップS47)。そして、再び選択情報の分類表示を行なうために選択情報の分類表示処理(D)へ進む(ステップS48)。

【0067】次に、図25乃至図29を参照して、本実施例に関する具体的な利用例を説明する。図25には、本実施例のスタート時点での情報処理装置の表示例を示してある。情報処理装置は図中の左側の広い表示領域21に情報単位集合を表示しており、その要素である情報単位は文書やアプリケーション、或いはプリンタなどの外部機器を参照するためのアイコンとなっている。また、図中の右側の領域22は選択情報リスト表示部であり、この例ではジョブ名と流入オブジェクト名の階層構造で選択情報が示されている。現在の状態は、利用者が4つの仕事をこのワークスペースで宣言しており、それは「Database特許」、「情報検索特許」、「月報資料」、「報告書執筆」である。例えば「Database特許」というジョブでは、「Doc-1.1」、「patent.1」、「Doc-3」、「test.1」、「Patent-2」の5つの文書を生成していることを示している。また、現在示されている仕事の場

の状態は、「月報資料」というジョブ名で「appendix」という文書を生成した後である。

【0068】例えば、この後、新しい特許原稿を書く仕事を始めようとする場合を考える。その特許は知識データベースに関する特許で、以前のジョブである「Database特許」に関する従来技術調査を利用できるとする。図26に示すように、選択情報としてジョブ名「Database特許」を選択することにより、ジョブ名「Database特許」の最新の状態に仕事の場の状態を戻すことができる。なお、「Database特許」のジョブで最後に「Patent-2」を生成した後の状態に戻ったことが、選択情報リスト22の表示により確認できる。

【0069】一方、「Database特許」のジョブでは2つの特許原稿を書いていたため、「Patent-2」の方ではなく、「patent.1」の方で使った技術調査が使いたいとする。その時は、図27に示すように、選択情報リスト22から「patent.1」を選択し、「Database特許」のジョブの途中の状態に仕事の場を戻すことができる。これにより、「Patent-2」に関する情報は仕事の場から排除することができる。これはつまり、仕事の場を論理的に分類しているのではなく、仕事の場の状態の変遷を記憶していることにより達成できる。この機能により、仕事の場の分岐を、後から異なる視点により切り直すことが可能になる。また、選択情報として利用者を名をいれれば、共同作業を行っていた仕事の場から、自分で行なった作業だけを切り出し、そこから新しい作業へと分岐させていくことも可能になる。

【0070】次に、その状態から新しい仕事を始める場合には、図28に示すように、例えば「New Job」というボタン23を押すことにより、ジョブ名の宣言のためのウィンドウ24がポップアップし、ジョブ名の宣言を促す。この例では、利用者は「知識DB特許」と入力している。そして「OK」を押すと、カレントの情報単位集合にジョブ名として「知識DB特許」が付けられる。その後、図29に示すように、例えば「Present-1」という文書を作成したとすると、選択情報リスト表示部22には、ジョブ名として「知識DB特許」、オブジェクト名として「Present-1」が加えられることになる。

【0071】本発明の特徴及び従来技術との違いを説明しておく。Roomsでは、まず「Database特許」のルームを作り、そこで仕事を行う。そして、次の仕事を行おうと思った場合には、「情報検索特許」というルームを作って、そこに新しい環境を構築していくことになる。ここで、特許原稿の執筆に必要なプリンタやエディタ、特許DBといったアプリケーション、或いは、共通して参照するドキュメントなどはどちらの仕事にも使うため、Roomsではコピーすることになる。

しかしながら、この作業は、必要な情報単位をルーム間で全てコピーする必要があるため、煩雑なものになる。そのため、複数の類似の業務、例えばこの例のような複数の特許原稿の執筆などは同じルームを使うことになりやすい。

【0072】これに対して、本発明では、ルームを新たに作るのとは違い、ある時点での状態に名前を付けてそれを呼び出す形をとっているため、類似のルームを構築する手間が省け、容易に仕事を細分化していくことができる。つまり、仕事を派生させると、前の状態がそのまま残っている。新しいジョブを宣言した直後は、ワークスペースは前のジョブの最終状態と同一の状態である。そこで、次のジョブに不要なものを片づけていき（つまり、要らないものは消去して）、必要な情報単位を収集していき、新しいジョブ環境を整えていくことになる。これを実世界のデスクに例えると、Roomsは正に仕事毎にたくさんのデスクを持つようなものである。これに対し、本発明は、仕事を行っているデスクの状態のスナップショットを随時取って仕事の名前を付けておき、その名前を宣言することによってデスクの状態を以前その仕事をしていた状態に戻すものである。

【0073】デスクには、当然どの仕事にも共通して使用するものもあるし、仕事によって意味付けの変わるものもある。これらの各情報単位の特徴をあらかじめ知ることは難しく、本発明では、仕事を進めながら自然と情報単位が分類されていくことが実現される。また、本発明では、「情報検索特許」のジョブで不要なものを片づけてしまった後、再び「Database特許」の仕事を行おうと思った際は、選択情報リストのジョブ名を指定するだけで前の状態が復元される。なお、本発明は、過去の状態を利用して、新しい状態を作ることを目的とするものであって、過去を復元することのみを目的としたものではない。もちろん、オブジェクト自体のバージョン管理も行うようにすれば、文書の編集状態をワークスペースの記憶時期に戻すことも可能であり、共同執筆などを支援する場合に効果がある。

【0074】上記した第1実施例には、例えば下記のような種々な変更を加えることができる。記憶指示部7は操作コマンド蓄積部9を持たず、情報単位集合記憶判断部8が利用者の記憶指示に従って記憶実行を行なうように構成してもよい。また、記憶指示部7は操作コマンド蓄積部9を持たず、情報単位集合記憶判断部8が、利用者による操作を監視して、情報単位集合の作成・削除の指示が行なわれたときに記憶実行を行なうように構成してもよい。また、情報単位集合表示部13は下地画像を用いて2次元或いは3次元的に表示してもよいし、情報単位の名前のアルファベット順或いは生成日時などでソートしてリスト表示してもかまわない。

【0075】また、情報単位集合の履歴記憶に関する実現方法としては、情報単位集合保持部14が保持する情

報単位集合と、情報単位集合保持履歴記憶部3に記憶されている最新の情報単位集合を常に一致させておき、情報単位集合の記憶を行なうタイミングで、履歴記憶部3内の情報単位集合をコピーし、コピーした情報単位集合に新しいidを与えて、それを情報単位集合保持部14に読み出すという方法でもよい。また、上記の実施例では、情報単位集合の履歴記憶に関する実現方法として、仕事を切り替えたタイミングで必ず直前の仕事の状態を保存するとしたが、必ず仕事を切り替えた直後の仕事の状態を保存するという方法でもよい。

【0076】また、情報単位記憶部14は情報単位のバージョンを管理することも考えられる。これは、既存のバージョン管理技術により容易に達成でき、この場合、情報単位記憶部14は情報単位の各バージョンの集合を更新日時とともに管理する。これにより、情報単位集合を復元した際に、情報単位の保持する情報もその情報単位集合を記憶した時点の状態に戻すことができる。また、オブジェクト管理部を設け、情報単位の参照先のオブジェクトのバージョンを管理することも考えられる。これは、既存のバージョン管理技術により容易に達成でき、この場合、オブジェクト管理部はオブジェクトの各バージョンの集合を更新日時とともに管理する。これにより、情報単位集合を復元した際に、情報単位の参照先のオブジェクトもその情報単位集合を記憶した時点の状態に戻すことができる。

【0077】また、記憶実行ルール指示部を設け、情報単位集合の記憶実行ルールを利用の局面にあわせてカスタマイズすることが考えられる。記憶実行のタイミングとしては、仕事の変化とみなすことができるタイミングがふさわしく、例えば、情報単位の属性を変えるタイミング、例えば情報単位の位置を変える、或いは情報単位に仕事の進行状況（例えば未着手・進行中・完了などのステータス）を表す情報が付けられたとすると、それを変更するタイミング等を用いることができる。また、記憶指示部7は操作コマンド蓄積部9を持たず、情報単位集合記憶判断部8が一定時間毎に情報単位集合を定期的に記憶することも考えられる。これにより、仕事の流れの概略を残すことができる。また、情報単位集合記憶判断部8は、情報単位を一定個数流入させたら情報単位集合を記憶するといった方法をとることも考えられる。これにより、一定作業の進行毎に履歴を残すこともできる。更に、適度に間引いた状況の保存ができ、仕事の粗密に影響されずに仕事の流れの概略を残すことができる。

【0078】また、情報単位集合の記憶判断に関する記憶実行ルールのチェックは、シーケンシャルに行なわれてもよいし、Rule-1を確認するときに他のルールが満たすかどうかを並行してチェックしてもよい。また、先に操作コマンド蓄積部9を参照して末尾の操作から逆引きし、満たす可能性のあるルールを絞り込んでも



よい。また、操作コマンド蓄積部9は、利用者による操作だけではなく、次に挙げるようなイベントを蓄積し、それに基づいて、情報単位集合記憶判断部8が記憶を行なうかの判断を行なってもよい。例えば、他のアプリケーションが起動した、或いは、他のアプリケーションがネットワークやプリンタなどの外部デバイスにアクセスしたといった同一コンピュータ上の他のアプリケーションのイベント。電子メールの到着或いは発信といったイベント。同一コンピュータ上で稼動しているワークフローツールにおける仕事の到着・発信といったイベント。その他、あらゆる他のアプリケーションの発生するイベントを取り込むことにより、仕事の切り替えのタイミングをより詳細に捉えることができる。

【0079】図30には、本発明の第2実施例に係る情報処理装置の構成を示してある。なお、前述した第1実施例と共通部分には同一符号を付して重複する説明は省略する。本実施例では、指定された複数の情報単位集合が保持する情報単位を集合演算することで合成し、新たな情報単位集合を生成する例を説明する。サーバ1は、第1実施例の構成に加え、複数の情報単位集合が保持する情報単位の集合演算を行なう情報単位集合合成部31と、情報単位集合が保持する情報単位のフィルタリングを行なう情報単位抽出部32とを有している。情報単位集合合成部31は、選択された複数の情報単位集合が保持する情報単位に対して「和集合」、「共通集合」、「差分」といった集合演算を行なう。情報単位抽出部32は、選択された情報単位集合が保持する情報単位に対して、その情報単位が保持する作成者・作成日時・コメント・起動アプリケーション・配置などの属性により各種フィルタリングをかけ、その結果として新たな情報単位集合を生成する。

【0080】クライアント11は、第1実施例の構成に加え、集合演算の指示を与える情報単位集合合成指示部35と、フィルタリングの指示を与える情報単位集合抽出指示部36とを有している。情報単位集合合成指示部35は、図32に示すように、合成するための情報単位集合を複数指定し、それに対して合成方法を指示するためのモジュールである。例えば、図31に示す「合成」ボタン25を押すことにより、図32に示すウィンドウ41が立ち上がる。そして、情報単位集合合成指示部35は、選択した情報単位集合に対して「和集合」42、「共通集合」43、「差分」44、「フィルタリング」45を指示する機能を有しており、利用者は、2つ以上の情報単位集合を指定し、指定した情報単位集合が保持する情報単位に対して「和集合」或いは「共通集合」の集合演算を行なう指示を与えることができる。また、利用者は、基準となる情報単位集合1つと、その他の1つ以上の情報単位集合を指定し、「差分」の集合演算を行なう指示を与えることができる。

【0081】情報単位集合抽出指示部36は、例えば図

31或いは図32に示す「フィルタリング」ボタン26、45を押すことによって、図33に示すウィンドウ51を呼び出す。利用者はこのウィンドウ51に条件を入力することにより、情報単位集合が保持する情報単位を抽出することができる。例えば、「作成者」に自分の利用者名を入れてフィルタリングすれば、カレントの情報単位集合から自分以外の利用者が作成した情報単位が全て削除される。また、図32に示す合成指示ウィンドウ41から呼び出された場合は、合成する情報単位をあらかじめフィルタ条件に絞って合成演算を行なうことになる。例えば、フィルタ条件として起動アプリケーションを自分の使っているエディタにすれば、エディタで参照可能な「文書」だけに絞って情報単位集合の合成を行なうことができる。

【0082】その他、「日時」として情報単位の作成日の範囲を指定する、「コメント」として指定するコメントを含む情報単位に絞る、「配置」として2次元空間上のある範囲に配置されている情報単位に絞る、といった条件で情報単位のフィルタリングを指示することができる。また、図33に示した例以外でもフィルタリングを行なうことはでき、例えば情報単位の参照するリソースの種類が「文書」なのか「アプリケーション」なのかといった属性、或いは参照するリソースの属性や、参照するリソースの含む文字列などによるフィルタリングも考えられる。

【0083】図34には、和集合、共通集合、差分の合成演算の概念を示してある。「和集合」は複数の仕事の統合に用い、「共通集合」は複数の仕事に共通の環境の抽出に用い、「差分」はある仕事に特有のリソースの抽出に用いることができる。和集合では、複数の情報単位集合のOR演算を行ない、集合 $A = \{a, b, c, d\}$ 、 $B = \{a, c, d, e, f\}$ ならば、 $A + B = \{a, b, c, d, e, f\}$ となる。また、共通集合では、AND演算を行い、 $A * B = \{a, c, d\}$ である。また、AとBの差分は、Aに含まれる情報単位からBに含まれる情報単位を差し引いたもので、 $A - B = \{b\}$ である。なお、Aに含まれていなく、Bに含まれているものは、この演算では無視される。

【0084】次に、図35乃至図38に示すフローチャートを参照して、情報単位集合の和集合、共通集合、差分、フィルタリングに関わる処理を説明する。図35には、和集合をとる場合の処理の手順を示してある。まず、図32に示した合成指示画面41でいくつかの情報単位集合が選択されて和集合ボタン42が押されると、選択された情報単位集合のidを取得する(ステップS50)。例えば、図32ではジョブ名が「Database特許」、「情報検索特許」及び「情報共有まとめ」が選択されて、それぞれ最新の情報単位集合が検索される。次いで、検索された各情報単位集合が保持する情報単位のidを情報単位集合—情報単位対応テーブルから



取得し(ステップS51)、それぞれの情報単位集合に関して情報単位idの集合を生成する。

【0085】そして、それらの情報単位のidをマージし、同一のものを削除するといった処理を行って、それぞれの情報単位の集合の和集合をとり、和集合に含まれる各情報単位をコピーして、それぞれに新しい情報単位idを付ける(ステップS52)。次いで、新しい情報単位集合を生成し(ステップS53)、情報単位集合—情報単位対応テーブルにステップS52でコピーしてidを付けた情報単位を全て加える(ステップS54)。この後、ワークスペース—情報単位集合対応テーブルに新しい情報単位集合のidを加え(ステップS55)、情報単位集合の再表示を行なう(ステップS56)。

【0086】図36には、共通集合をとる場合の処理の手順を示してある。まず、図32に示した合成指示画面41で、いくつかの情報単位集合が選択されて共通集合ボタン43が押されると、和集合の例と同様に、選択された情報単位集合のidを取得し(ステップS60)、各情報単位集合が保持する情報単位のidを情報単位集合—情報単位対応テーブルから取得して(ステップS61)、それぞれの集合を作る。次いで、全ての情報単位のAND集合をとり、そのAND集合の各情報単位をコピーし、それぞれに新しい情報単位idを付ける(ステップS62)。

【0087】そして、新しい情報単位集合を生成し(ステップS63)、情報単位集合—情報単位対応テーブルに、ステップS62でコピーしてidを付けた情報単位を全て加える(ステップS64)。この後、ワークスペース—情報単位集合対応テーブルに新しい情報単位集合のidを加え(ステップS65)、情報単位集合の再表示を行なう(ステップS66)。

【0088】図37には、差分をとる場合の処理の手順を示してある。まず、図32に示した合成指示画面41で、いくつかの情報単位集合が選択されて差分ボタン44が押されると、和集合の例と同様に、選択された情報単位集合のidを取得し(ステップS70)、各情報単位集合が保持する情報単位のidを情報単位集合—情報単位対応テーブルから取得する(ステップS71)。次いで、合成演算の基準となる情報単位集合の入力を利用者に促す(ステップS72)。例えば、「基準となる情報単位集合はどれですか?」といったメッセージとともに、チェックされた全ての情報単位集合をディスプレイ画面に表示出力する。図32の例では、「Database特許」、「情報検索特許」、「情報共有まとめ」の3つの情報単位集合が選択され、そのうちの1つを利用者が選択してそれを基準とした差分をとることになる。

【0089】例えば、「Database特許」を基準として選んだときには(ステップS73)、基準となる情報単位集合(つまり「Database特許」の最新の情報単位集合)が保持する情報単位idのリストか

ら、他の情報単位集合(つまり「情報検索特許」及び「情報共有まとめ」のそれぞれ最新の情報単位集合)が保持する情報単位idのリストに含まれているものを除く処理を行う。そして、残った情報単位を全てコピーし、それぞれに新しい情報単位idを付ける(ステップS74)。次に、新しい情報単位集合を生成し(ステップS75)、情報単位集合—情報単位対応テーブルに、ステップS74でコピーしてidを付けた情報単位を全て加える(ステップS76)。この後、ワークスペース—情報単位集合対応テーブルに新しい情報単位集合のidを加え(ステップS77)、情報単位集合の再表示を行なう(ステップS78)。

【0090】次に、フィルタリングを行う場合の処理の手順を説明する。図33に示したフィルタ条件入力画面51は、図31或いは図32の画面においてフィルタリングボタン26、45が押されることによって表示される。この呼び出し方の違いによって、フィルタリングの機能は次の2つに分かれる。

(1) 図31に示した情報単位集合表示画面から呼び出された場合で、カレント情報単位集合に対してフィルタリングを行なう機能、(2) 図32に示した合成指示画面から呼び出された場合で、前記のように合成を行なうときに合成対象の情報単位をフィルタ条件で制限する機能。

【0091】図38には、前者の(1)に関するフィルタリング処理の手順を示してある。図33に示したフィルタ条件入力画面51でOKボタンが押されると、指示されたフィルタ条件を獲得する(ステップS80)。次いで、ワークスペーステーブルから、カレント情報単位集合のidを取得し(ステップS81)、カレント情報単位集合が保持する情報単位のidを情報単位集合—情報単位対応テーブルから取得する(ステップS82)。そして、各情報単位に対してフィルタ条件をかける(ステップS83)。例えば、フィルタ条件が「作成者=Aoki」であれば、作成者idが利用人名Aokiである情報単位のみを選択する。

【0092】そして、選択された情報単位を全てコピーして、それぞれに新しい情報単位idを付け(ステップS84)、新しい情報単位集合を生成し(ステップS85)、情報単位集合—情報単位対応テーブルに、ステップS84でコピーしてidを付けた情報単位を全て加える(ステップS86)。この後、ワークスペース—情報単位集合対応テーブルに新しい情報単位集合のidを加え(ステップS87)、情報単位集合の再表示を行なう(ステップS88)。

【0093】一方、後者の(2)に関しては、和集合・共通集合・差分の合成演算を行なうときに、合成して得られた集合に対して(1)と同じフィルタリングを行なうことにより演算が達成される。

【0094】次に、「和集合(重ね合わせ)」、「共通

集合（掛け合わせ）」、「差分」の演算と、「フィルタリング」の組み合わせの具体的な例を用いて効用を説明する。「和集合（重ね合わせ）」は、複数の情報単位集合が保持する情報単位に対して和集合をとる演算である。複数の仕事を統合する場合に、これを用いると効果がある。例えば、2つの特許原稿を別の作業環境で書いており、後からそれらを1つの特許原稿にしようと思った場合、和集合をとることにより、両方の状態の保持する情報がマージされ、新たな環境が作られる。具体例で述べると、2つの特許原稿を別のジョブで執筆していて、それらを1つの特許原稿にして書こうと思った場合、一方の特許原稿のジョブと他方の特許原稿のジョブのそれぞれ最新の情報単位集合を重ね合わせる。これにより、両方のジョブに共通の情報単位と、各々のジョブに固有の情報単位がマージされて表示される。

【0095】例えば、情報単位集合1＝{プリンタ、エディタ、参照特許A、参照特許B、参照特許C、執筆中文書1、執筆中文書2}と、情報単位集合2＝{プリンタ、エディタ、グラフィックツール、参照特許A、参照特許D、参照特許E、執筆中文書3、執筆中文書4、執筆中文書5}という2つの情報単位集合があり、それぞれで異なる特許原稿を執筆していたが、あるときに一つの特許にしようと思い立った場合、重ね合わせることができる。重ね合わせた情報単位集合は、情報単位集合 $(1+2) = \{\text{プリンタ、エディタ、グラフィックツール、参照特許A、参照特許B、参照特許C、参照特許D、参照特許E、執筆中文書1、執筆中文書2、執筆中文書3、執筆中文書4、執筆中文書5}\}$ となる。

【0096】この機能を用いることにより、どの情報単位が共通に用いられており、どの情報単位は固有のものか、ということ意識せずに情報単位集合の合成が可能になる。このとき、フィルタリングの条件として「アプリケーション」を選択しておけば、情報単位集合 $(1+2) * (\text{「アプリケーション」}) = \{\text{プリンタ、エディタ、グラフィックツール}\}$ となり、特許原稿で使うアプリケーションをすべて抽出することができる。なお、この例では2つの情報単位集合を重ね合わせる例を示したが、例えば、複数のスペースで（複数人で）それぞれアイデアを練っており、それを合成して提案書を作るとき、或いは、複数のスペースで（複数人で）それぞれ文献の収集とそれに対するコメント付けを行っており、それらをまとめるとき等では、もっと多くの情報単位集合の合成が必要になり、この機能の効果がより際立つ。

【0097】「共通集合（掛け合わせ）」は、複数のスナップに共通の二次情報単位を抽出する機能である。例えば、いくつかのジョブで特許原稿を書いていたとする。それらのジョブのスナップを掛け合わせることで、特許原稿を書くために必要な二次情報単位が抽出される。このようにして特許原稿を書くための基本的な環境を得ることができる。前述の情報単位集合1と2の例

で掛け合わせを行うと、情報単位集合 $(1*2) = \{\text{プリンタ、エディタ、参照特許A}\}$ という新たな情報単位集合が抽出される。これを新たな特許原稿を書くための情報単位集合のひな型とすることができる。例えば、特許原稿を多く書いている人のワークスペースを参照させてもらう場合を考える。他人の使っているワークスペースなので、どの環境が自分のやりたいことにあったものかわかりにくい。そこで、例えば特許原稿に関連しそうなジョブ名をいくつか選択し、それらの掛け合わせを行うことで、その人がいつも特許原稿を書くときに使っている最低限の環境、或いは常に参照している文書は何か、といったノウハウを抽出することができる。さらに、そのノウハウを反映させた情報単位集合を生成し、そこで仕事を始めることができる。

【0098】「差分」は、掛け合わせと逆に、スナップに特有の情報を抽出するための機能である。ある仕事を行っているスナップから、基本状態のスナップを引けば、結果としてその仕事に固有の情報を切り出すことができる。例えば、情報単位集合3＝{プリンタ、メール、エディタ、データベース、メモ帳、時計、アイデアノート、スケジュール帳}という情報単位集合を通常状態として用いていたとする。そして、情報単位集合1で執筆中の特許原稿に固有の情報単位を抽出すると、情報単位集合 $(1-3) = \{\text{参照特許A、参照特許B、参照特許C、執筆中文書1、執筆中文書2}\}$ が選択され、これらがこの特許原稿を書くために用いられている情報単位であることがわかる。この機能を用いれば、類似の仕事複数のジョブで行っていた場合に、どの情報単位が必要なのかわからなくなってきた場合など、差分をとることでそのジョブのエッセンスの抽出を行い、整理を行うことができる。また、例えばある発明の特許原稿を書いている作業環境があり、仮に同じテーマの報告書を書いている作業環境が他にあったとすると、それらの作業環境の差分をとることにより、特許原稿を書くための作業環境に特有の情報が何かを知ることができる。

【0099】なお、これらの演算を行う際に情報単位同士のマッチングを行っているが、このマッチングにおいて、その情報単位が同じであるかどうかの決定方法は、実施例で示した情報単位のidの同一性の他に、情報単位名の同一性、参照オブジェクトの同一性に関して、或いはそれら双方のマッチングを行うことが考えられる。この同一性の基準は、いくつかの選択肢を利用者に与え、利用者自身がマッチング方法を選択することも考えられる。

【0100】ここで、従来技術と本発明との相違を説明しておく。これまでのフォルダやRoomsなどのシステムでは、利用者があらかじめ仕事に合わせて情報を分類する必要があった。これに対し、本発明では、仕事を進めながら、仕事毎に情報の整理を行うことができる。そのため、仕事に合わせた柔軟な情報の体系化が可能に

なる。これが可能になるのは、Roomsでは「複数の仕事の場の、現在の状態」を保持しているのに対し、本発明では「(1つの)現在の仕事の場の状態」と「複数の過去の仕事の場の状態」を区別して記憶する。これにより、以下に挙げる2つの特長が得られる。

【0101】(1)現在の仕事の場の状態をどんな仕事と関連づけて記憶しておくか、ということに自由度がある。そのため、仕事をある程度進めてから、これまでに行った仕事はこれまでとは違う仕事と関連付けた選択情報をつけて保存しよう、ということが可能になる。

(2)過去の仕事の場の状態に複数の属性を付与しておくことにより、現在の仕事の場の状態へ読み出し可能な過去の状態の選択肢を変化させることができる。例えば、Roomsでは「ルーム」の名前でしか状態を区別・保存できないが、本発明では、仕事の場の状態に「仕事の名前」「利用者名」といった2つの属性を付けておけば、「仕事の名前」だけで指定すればその仕事の名前の最新の仕事の場の状態を現在の状態として使え、「仕事の名前」「利用者名」の両方を指定して例えば自分(或いは他の誰か)が以前その仕事を行っていた状態を呼び出すことも可能になる。

【0102】また、Triggらのヒストリーワークスペース特許との最大の違いは、仕事の履歴を残しておく、その仕事の場を再利用することができることである。Triggらのヒストリーワークスペースでは、操作履歴がリストとして新たなワークスペースを構成する。これにより、利用者は、以前行った仕事のプロセスを参考にしたり、その履歴から情報を見つけたといった利用が可能である。つまり、ヒストリーワークスペースは仕事の操作履歴をリストにすることにより、ワークスペース間の関係を利用者に示すものである。しかしながら、ヒストリーワークスペースではワークスペース間の呼出関係が保存されるが、本発明では保存する内容が異なり、ワークスペース上で行われたオブジェクト操作とワークスペースの状態が保存される。つまり、本発明はヒストリーワークスペースよりもより詳細な情報を保存しておき、それにより、より詳細な場面再現を可能にしている。またさらに、これにより、呼び出した過去の状態を利用して新たな仕事を始めるといったことが可能になる。

【0103】また、進行状態の保存や履歴の再利用に関する発明は、履歴利用の観点がこれまでのものとは異なる。履歴情報を利用した新たな仕事環境の生成としては、オペレーションの再利用のみで、そこから抽出できる仕事の体系やノウハウを充分活用していない。これに対し、本発明では履歴を単なる過去の一場面とみなすのではなく、過去の各場面を一つ一つの集合とみなすことにより場面の保持するオブジェクトのグループでオブジェクトを分類する。また、本発明は、関連情報やノウハウの参照・再利用ができ、同じ仕事を行っているワーク

スペースの複数の状態を保存しておくことにより、過去の仕事の状態や関連情報を参照したり、過去の仕事の状態に戻って新たな仕事を始めることができる。つまり、ワークスペース上で行われた履歴が蓄積されるため、ある仕事に必要な情報は何か、ある仕事でよく使われる情報は何か、或いはある仕事に固有の情報は何か、といった仕事のノウハウを抽出・再利用することができる。

【0104】図39には、本発明の第3実施例に係る情報処理装置の構成を示してある。なお、本実施例は主に請求項24～請求項37に記載した発明に該当している。本実施例の情報処理装置には、複数の情報単位集合を更に集合化したプレース(Place)を複数記憶する蓄積部61と、表示対象となる複数の情報単位集合をその選択情報と共に保持する現在状態保持部62と、現在状態保持部62に保持されている情報単位集合の内の利用者によって入力された選択情報に対応するものをディスプレイ画面に表示出力する現在状態表示部63と、予め設定された選択情報についての抽出規則に対応する情報単位集合を蓄積部61から抽出する関連状態抽出部64と、当該抽出された情報単位集合を保持する関連状態保持部65と、当該抽出された情報単位集合の選択情報のリストをディスプレイ画面に表示出力するリスト表示部66と、指定された情報単位集合を関連状態保持部65から読み出して現在状態保持部62に保持させる関連状態獲得部67、現在状態保持部62に保持されている情報単位集合を蓄積部61に保存するか否かを予め設定した規則に従って判断するとともに保存処理を行う保存判断部68と、蓄積部61に保持されているPlaceの内の利用者によって選択されたものを現在状態保持部62に保持させる現在状態導入部69と、が備えられている。

【0105】ここで、本実施例では、前述した情報単位はメディエータ(Mediator)と称するオブジェクトとされ、また、情報単位集合はスナップ(Snap)と称するオブジェクトとされており、更に、幾つかのSnapはプレース(Place)と称するオブジェクトとしてまとめられている。また、本実施例の蓄積部61は前述した情報単位集合保持履歴記憶部3にほぼ相当し、関連状態導入部67は前述した情報単位集合読出部5にほぼ相当する。また、本実施例の現在状態保持部62は前述した情報単位集合保持部14にほぼ相当しており、更に本実施例では、情報単位集合保持部14の機能として現在状態表示部63及び現在状態獲得部69が備えられているものとして構成されている。また、本実施例の保存判断部68は前述した記憶指示部8にほぼ相当し、更に本実施例では、保存判断部68が情報単位集合(Snap)を蓄積部61へ記憶させる蓄積手段の機能も有している構成としてある。また、関連状態抽出部64、関連状態保持部65、リスト表示部66、及び、後述する関連情報指定手段は前述した情報単位集合選択

部4にほぼ相当し、利用者によって指示された選択情報に対応する情報単位集合(Snap)を選択する。

【0106】なお、本実施例の情報処理装置は、蓄積部61のみがサーバ側に置かれ、他のモジュール62～70はクライアント側に置かれたサーバ・クライアント形式で構成されており、サーバとクライアントとのいずれもワークステーションあるいはパーソナルコンピュータで構成されている。ただし、本発明においては、サーバ・クライアントに分けた形態をとる必要はなく、例えば、蓄積部61もクライアント側に置かれて、複数のクライアント間で蓄積部61が保持する情報を交換する方法をとってもよい。

【0107】上記した蓄積部61はリレーショナルデータベースであり、図40(a)～(f)に示すようなPlace、Snap、Job、Mediator、User、IDMaxといったデータベーススキーマを有している。Placeは幾つかの仕事をする場所を代表するオブジェクトであり、Placeの識別子(placeID)、生成日時(createDate)、生成したユーザのid(user)、Place(場所)の名前(placeName)、その場所で行っている複数の作業環境(Snap)の最新の状態の集合(listSnap)、を組で有している。例えば、Place = {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 'Nomura home', '1,2,6'}といった構成となっている。

【0108】Snapは作業環境の状態を示すオブジェクトであり、Snapの識別子(snapID)、生成日時(createDate)、生成したユーザのid(user)、そのSnapで行っている仕事を示すJobのid(myJob)、そのSnapが作られたPlaceのid(myPlace)、そのSnapの保持する情報単位(mediator)の集合(mediators)、を組で有している。例えば、Snap = {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, '1,3,4'}といった構成となっている。

【0109】Jobは仕事の種類を表すオブジェクトであり、Snapを選択するためのラベルとして機能する。JobはJob(仕事)の識別子(jobID)、生成日時(createDate)、生成したユーザのid(user)、その仕事の名前(jobName)、を組で有しており、例えば、Job = {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 'CSCW survey'}といった構成となっている。なお、仕事の名前が同じであっても、別のオブジェクトであれば別の仕事として判別される。そのため、仕事を複数のPlaceで共同で行うためには、Jobのコピーを行う必要があり、本実施例ではこのコピーをSnapをPlaceに導入することにより実現している。

【0110】Mediatorは情報単位を表すオブジェクトであり、Mediatorの識別子(media

torID)、生成日時(createDate)、生成したユーザのid(user)、Mediatorの名前(myName)、Mediatorの指すURL(Uniform Resource Locator)などのリファレンス先アドレス(myLocation)、Mediatorの保持するメモ(myMemo)、Mediatorの表示されるX座標(myMX)、Mediatorの表示されるY座標(myMY)、を組で有している。例えば、Mediator = {1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 'report draft', 'http://abc.www.abcdef.co.jp/', 'This is the first draft of the report. (nomura)', 1, 3}といった構成となっている。

【0111】Userはユーザを表すオブジェクトであり、Userの識別子(userID)、生成日時(createDate)、生成したユーザ名(userName)、ユーザのパスワード(password)、ホームPlaceの識別子(homePlace)、を組で有している。例えば、User = {1111, 'Wed Jul 10 15:00:00 JST 1996', 'nomura', '\*\*\*\*\*', 1}といった構成となっている。ここで、ホームPlaceとは、そのユーザが主に利用するPlaceのことであり、例えばログイン時に開かれたり、Mediatorがシステム外から電子メール等によりそのユーザに配送されたときのMediatorの配送先となるPlaceである。

【0112】IDMaxは複数のクライアントが同時にPlace、Snap、Job、Mediatorのいずれかを生成した場合にもデータベース内でid(identifier:オブジェクトのユニークネスを示すラベルという意味で使用)の一貫性がとれるようにするテーブルであり、例えば、IDMax = {3, 12, 7, 23}といった構成となっている。

【0113】この一貫性の保持について説明しておく、クライアントが例えばMediatorを1つ生成した場合、まず、そのMediatorのidは仮のものに設定しておく処理を行う。そもそもidの役割は、データベースから情報を読み込んでオブジェクトを生成する際にオブジェクト間の関係を復元するためのものであり、クライアントの同一セッション内ではMediator等の参照はオブジェクト単位で行われるのでidは必要なく、idはデータベースに格納するときに割り当てられている必要があるだけである。そこで、データベースへの保存命令がユーザから行われた際に、クライアントはオープンしているPlace内の全てのMediatorをサーチし、新たに生成されたMediatorの数を知り、IDMaxの"mediator"の値(この例では23)を獲得し、これに対して新たに生成されたMediatorの数を足した数で更新する。このようにidの予約を行ってから実際にidを割り当

てて保存することにより、複数のクライアントで同時にオブジェクトを生成してもidのバッティングは起こらない。なお、idの決定方法はこのような方法に限る必要はなく、生成日時やユーザidなどのパラメータを組み合わせて動的に計算するなどの方法をとってもよい。

【0114】上記したPlace、Snap、Job、Mediatorから成るデータは、現在状態保持部62には図41(a)に示すようなモデル構成で保持され、蓄積部61には図41(b)に示すようなモデル構成で保持されている。すなわち、現在状態保持部62にはPlace単位でデータが保持されており、このPlace単位のデータがユーザに提示される。ここで、Placeは現在その場所で進められている複数の仕事の状態を保持しており、データ構造としては、複数のSnapを保持あるいはポインタによって指示している。なお、各SnapはそれぞれJobを指しており、結果として、Placeは複数のJobを指すことにもなっている。一方、蓄積部61(データベース)内では、Place、Snap、Job、Mediatorはそれぞれ別々に管理されており、お互いのidを持ち合っており、結び合わされている。

【0115】上記した現在状態表示部63は、本実施例では図42に示すような表示画面を有しており、表示出力とともにユーザからの入力を受け付けるインタフェースとなっている。当該画面の各部は次のような機能を有している。ログインフィールド72にログイン名を入力後、ログインボタン71を押すことでログインが行われ、ログインが行われるとそのユーザのホームPlaceがPlace選択メニュー74のデフォルトとして表示される。また、オープンボタン73が押されると、Place選択メニュー74により選択されたPlaceが開かれ、そのPlaceの指すJobがジョブリストに表示され、ジョブリストの先頭のジョブを指すSnapが保持するMediatorの集合がスナップビューに表示される。Place選択メニュー74には選択し得る全てのPlaceのリストが表示され、ユーザはこのうちの1つを選択することで、任意のPlaceを画面上に開くことができる。

【0116】Saveボタン75が押されると、Placeを開いてから更新された全ての情報をデータベース61に書き出して保存する指示が与えられる。また、CallSnapボタン76が押されると、図44に示すウィンドウが呼び出されて画面表示され、このウィンドウにおいてデータベース61に格納されたSnapを抽出するための指示がユーザによって与えられる。ViewPileボタン77が押されると、図43に示す関連状態ウィンドウが呼び出されて画面表示され、このウィンドウによって関連状態保持部65が保持しているSnap集合の属性がリスト表示される。

【0117】Combineボタン78が押されると、

後述する図59に示すようなウィンドウが呼び出されて画面表示される。なお、このウィンドウにおいて複数のSnapを合成させる指示がユーザから入力されると、後述する第4実施例で説明するように、これに基づいて状態合成部により所定の集合演算処理が行われる。Newボタン79が押されると、Mediatorを新規生成させる処理が行われ、この新規生成処理では、Mediator名フィールドとURLフィールドに書かれた文字列を読み取り、それを新規Mediatorの属性とする処理が行われる。Cutボタン80が押されるとMediatorを現在処理対象としているSnapから削除し、さらに、Copyボタン81が押されると当該Mediatorをコピーバッファに保持させる処理が行われる。

【0118】Pasteボタン82が押されると、CutあるいはCopyしたMediatorをペーストする処理が行われ、CutあるいはCopyされたMediatorは元あったX、Y座標と同じ座標にPasteされる。なおもし、その座標に既に他のMediatorが存在した場合は、その右隣にMediatorが配置され、この位置にも他のMediatorが存在した場合には更に右隣に配置するといった処理が繰り返行われ、この結果として、表示画面の右端まで達してしまった場合には、Y座標を1つ繰り上げると共にX座標を左端に戻して同じ作業を繰り返し、空きがある位置に配置する処理が行われる。スナップビュー83はMediatorをアイコン84で表示する領域であり、この領域にはジョブリスト89で選択されているJobを指すSnapが保持しているMediatorが表示される。これら表示されたMediatorはマウスなどのポインティングデバイスにより、選択あるいはメモやリファレンスの参照が行われる。また、Mediatorの移動はポインティングデバイスのドラッグアンドドロップにより実行される。

【0119】Mediatorアイコン84はMediatorを表すアイコンであり、リファレンスの種類によりアイコンの表示が変えられている。例えばWindows(Microsoft社の登録商標)では".txt"などのフォーマット毎に決まった拡張子が割り当てられており、フォーマット毎に異なるアイコンを表示することができる。また、URLの文字列をパースして社内のURLであれば社内文書アイコン、社外のURLであれば社外文書アイコンといった区別も可能となっている。Mediator名フィールド85はスナップビュー83において選択されたMediatorの名前が表示される領域であり、URLフィールド86はスナップビュー83において選択されたMediatorのURLが表示される領域である。

【0120】NewJobボタン87が押されると、NewJob名フィールド88にユーザによって入力され

た名前（文字列）の新規Jobを生成する処理が行われる。なお、この新規Jobを指すSnapは、NewJobボタン87を押した時に選択表示されていたSnapをコピーして生成される。このため、仕事が派生した場合、あるいは他人の仕事を受け継ぐ場合など、以前のJobの作業環境をそのままに別の作業環境を生成することが可能となっている。ジョブリスト89は画面に開かれているPlaceの指す全てのJob名がリストとして表示される領域であり、この領域においてユーザがマウスなどのポインティングデバイスにより1つのJobを選択すると、スナップビュー83に表示されるSnapを切り替えることができる。なお、このような選択により、ユーザは複数の仕事を並行して行うことが可能となっている。DeleteSnapボタン90が押されると、ジョブリスト89で選択されているJobを指すSnapをこのPlaceから削除する命令がなされ、ImportSnapボタン91が押されると、図43に示す関連状態リストウィンドウで選択されている1つの関連状態SnapをこのPlaceに導入する命令が出される。

【0121】上記した関連状態保持部65は、関連状態としてSnapの集合を保持しており、関連状態抽出部64によりデータベース61からSnapが検索されると、これらのSnapを次々に集合に加えて保持する。ここで、本実施例ではSnapのオブジェクトを保持しているので、リスト表示部66においてユーザがポインティングデバイスによりその1つを指示すると、当該Snapの保持するMediatorの集合がスナップビュー89に表示される。また、後述するように保存判断部68により、現在状態保持部65で更新される直前の状態を保存するという判断が行われた場合には、そのSnapは関連状態保持部65に移動させられ、“isModifiedFlag”という更新されたことを示すフラグを立てておき、Saveボタン75が押されたときに一度にデータベース61に格納する処理が行われる。

【0122】リスト表示部66は図43に示すような画面表示を行い、この画面に関連状態保持部65に保持されているSnap集合の属性（選択情報）をリスト表示する。すなわち、当該画面のリスト表示部101はSnap集合の選択情報がリスト表示される領域であり、ユーザがこのリストのいずれか1つの項をマウスなどで選択することにより、選択されたSnapを図42に示したスナップビュー83に表示させることができる。なお、本例では、Snap集合の選択情報として、生成したユーザ名、ジョブ名、生成日時を表示している。また、Clearボタン102が押されると、関連状態保持部65が保持し且つリスト表示部66に表示されている全てのSnapを削除する処理が行われ、また、Closeボタン103が押されると、当該ウィンドウ（図43）をユーザから不可視の状態にする処理が行われ

る。なお、このようにCloseしても関連状態保持部65の保持内容はリセットされない。

【0123】関連状態導入部67は、関連状態保持部65が保持しているSnapのうちの1つをユーザがリスト表示部66に表示されたSnapリストから選択し、現在状態表示部63の画面のImportSnapボタン91を押すことによって起動される。そして、関連状態導入部67は選択されたSnapを後述するように現在状態保持部62に保持させる処理を行い、これによって、現在状態として対象としているSnapを変更することを可能ならしめる。保存判断部68は、現在状態保持部62が保持するSnapの指すMediator集合の要素（個々のMediator）が増加あるいは減少する、あるいは、ユーザがMediatorの参照あるいはコピーなどの指示を与える、といったことをきっかけに、Snapの内容が変化する直前の状態を保存するかどうかの判断を行う。なお、この判断によって保存すべき時には、内容が変化する直前のSnapがデータベース61に保存され、過去の仕事の復元等を可能ならしめる。

【0124】現在状態獲得部69は、Placeを画面に開いたときに、そのPlaceの指す全てのSnapをデータベース61から読み出し、さらに各Snapの指すJobを読み出し、さらに各Snapの指すMediatorを全て読み出し、それらをオブジェクトとして生成させる処理を行う。また、現在状態獲得部69は、この生成したオブジェクトを現在状態保持部62に保持させる処理を行い、これによって、現在状態として対象としているSnapを表示部63に表示出力することを可能ならしめる。

【0125】関連状態抽出部64は、データベース61に記憶されているデータの中からユーザが指定した選択情報に合致するものを抽出し、当該データを関連状態保持部65に保持させる。この抽出処理を実行するために、関連状態抽出部64は、図44に示すようなウィンドウ画面表示を行い、当該画面においてユーザは関連状態抽出のための検索指示を行う。当該画面の各部は次のような機能を有している。Place選択リスト部110は、関連状態抽出のための検索条件として、Place名をリスト表示する領域であり、ユーザはリスト中の所望のPlaceをマウス等の指定手段によって選択する。

【0126】Job選択リスト部111は、関連状態抽出のための検索条件として、Job名をリスト表示する領域であり、ユーザはリスト中の所望のJobをマウス等の指定手段によって選択する。なお、本実施例では、CallSnapsボタン76を押したときに画面上に開かれているPlaceが指しているSnapの指すJobのみがリストに表示される。ここで、他のPlaceのJobもリストにしても構わないが、その場

合、表示されるJobの数が非常に多くなるが、現在開いているPlaceのJobを優先して上位に表示するなど行えばよい。

【0127】User選択リスト部112は、関連状態抽出のための検索条件として、User名をリスト表示する領域であり、ユーザはリスト中の所望のUserをマウス等の指定手段によって選択する。Mediator名入力フィールド113は、関連状態抽出のための検索条件として、Mediatorの名前を入力するフィールドであり、ユーザは所望のMediator名をキーボード等の指定手段によって入力する。

【0128】Date(From)入力フィールド114は、関連状態抽出のための検索条件として、Snapの生成日時の始まりを入力するフィールドであり、ユーザは所望の日時をキーボード等の指定手段によって入力する。Date(To)入力フィールド115は、関連状態抽出のための検索条件として、Snapの生成日時の終わりを入力するフィールドであり、ユーザは所望の日時をキーボード等の指定手段によって入力する。したがって、Date(From)入力フィールド114とDate(To)入力フィールド115とに入力された日時の間(期間)に生成されたSnapが、関連状態抽出部64によって検索されることになる。

【0129】Retrieveボタン116が押されると、上記の条件(選択情報)によってデータベース61からSnapの検索が開始され、Closeボタン117が押されると、当該ウィンドウが閉じられる。SharedJobSnapsボタン118が押されると、現在開かれているPlaceの指すJobに関して、同じJobを共有している他ユーザの当該Jobを指す最新のSnapを検索してくる処理が行われる。なお、この処理に関して詳しくは後述する。MyPastSnapsボタン119が押されると、現在開かれているPlaceの指すJob以外で、過去に当該Placeで行われた全てのJobに関して、各Jobを指す最新のSnapを検索してくる処理が行われる。なお、この処理に関して詳しくは後述する。

【0130】次に、本実施例に係る情報処理装置の動作を説明する。まず、作業環境(Snap)をデータベース61に保存するタイミング(つまり保存判断部68の動作)について説明する。図45に示すように、仕事を始めると(Job宣言)、情報収集が行われて、作業環境(Snap)には様々な情報(Mediator)が集められてくる。一方、作業環境から情報が消去される場合もあり、このような状況は机を片づけるときの如く仕事の切り替えのタイミングと考えることができる。

【0131】そこで、Snap上のMediatorが増加から減少に転じる際に、この変化前のSnapをデータベース61に保存することを行い、これにより、仕事の切れ目の状態を保存して再利用することを可能にし

ている。また、この保存についてのアルゴリズムを採用することにより、全てのMediatorが消失しないことを保証することができ、これによって、ユーザはその時点で不要になった情報を迷わず作業環境から削除することができるようになる。この結果、後で使うかもしれないから取っておこうと思う必要がなくなり、ユーザによる作業負担が軽減する。

【0132】この保存判断部68による保存タイミングに関する処理を、図46に示すフローチャートを参照して更に詳しく説明する。Place(プレース)の各Snapは、そのSnapが既に保存されたかどうかを示す"alreadyPiledFlag"、そのSnapが保持するMediatorに変化があったかどうかを示す"isModifiedFlag"、そのSnapが新たに作られてから保存されていないかどうかを示す"isNewFlag"という3つのFlagパラメータを有しており、保存判断部68は現在状態保持部62に保持されている全てのSnapについてのこれらパラメータを管理している(ステップS90)。なお、Placeをオープンした(開いた)直後の各Snapは、alreadyPiledFlag=False、isModifiedFlag=False、isNewFlag=False、といった状態のFlagパラメータを有している。

【0133】次いで、SnapにMediatorが追加されたかを判断し(ステップS91)、SnapにMediatorが追加された場合には、そのSnapのFlagパラメータを、alreadyPiledFlag=False、isModifiedFlag=True、に変更する(ステップS92)。つまり、alreadyPiledFlag=Trueだったとしても、またMediatorが追加されることにより、既に保存された状態ではないということが表現される。一方、追加されていない場合には、Mediatorが削除されたかを判断し(ステップS93)、削除された場合には、そのSnapのFlagパラメータを、isModifiedFlag=True、に変更する(ステップS94)。なお、この場合には、alreadyPiledFlagは変更しない。

【0134】このようにMediatorの削除が行われた後に、Snapの保存を行うかどうかの判断を行い、"alreadyPiledFlag"がTrueであるか否かを判断する(ステップS95)。この結果、Trueである場合には、既に保存が成されているので保存処理を行わない一方、Falseである場合には、Mediatorを削除する直前のSnapの状態を保存する必要があるので、そのSnapをコピーし(ステップS96)、コピーしたSnapをデータベース61へ格納する(ステップS97)。なお、このデータベース61に格納されたSnapは関連状態抽出部64の処理によって関連状態保持部65に保持されることとなる。また、上記のコピー処理に際して(ステップS96)、コピーする直前のSnapのFlagパラメータは、alreadyPiledFlag=True、isModifiedFlag=True、isNewFlag=True、に設定される。



【0135】次に、更新されたデータを保存判断部68がデータベース61へ格納する処理を、図47に示すフローチャートを参照して説明する。Saveボタン75が押されてデータベース格納命令を受けると(ステップS100)、まず現在オープンされているPlaceの指すSnapの集合を獲得する(ステップS101)。そして、各スナップを順にループで回し(ステップS102)、それぞれのSnapの" isModifiedFlag"をチェックする(ステップS103)。この結果、Falseである場合にはそのままループチェックを続行する一方、Trueである場合には、さらに" isNewFlag"をチェックする(ステップS104)。

【0136】この結果、" isNewFlag" がFalseである場合には、そのSnapに関してデータベース61を更新する一方(ステップS105)、" isNewFlag" がTrueである場合には、新規Snapであるので、データベース61にそのSnapを追加する(ステップS106)。上記の一連の処理を全てのSnapについてのチェックが終わるまで、ループで繰り返して行い(ステップS107)、更には、現在オープンされているPlace、Job、Mediatorについての同様にチェック及び保存処理を行う(ステップS108)。なお、保存を行うときのオブジェクトのidは、IDMaxのスキーマに関して前述したような処理を行う。

【0137】次に、関連状態抽出部64が行う検索処理を説明する。仕事を遂行している状況において自分と関連のある仕事の作業環境を抽出する必要がある場合があり、図48にはその一例として、ユーザnomuraがJob1、Job2、Job3という3つのJobを現在状態に持っている場面で、関連Jobの検索指示を出した場合を示してある。この指示を出すことによって、Job1、Job2、Job3の各Jobに関して他のユーザの最新のSnapが関連状態抽出部64によってデータベース61から検索される。そして、検索されたSnapが関連状態保持部65に保持され、リスト表示部66にそのリストが表示される。

【0138】この抽出処理を図49に示すフローチャートを参照して更に詳しく説明する。まず、SharedJobSnapsボタン118ボタンが押されると(ステップS110)、現在開いているPlace(すなわち、theOpeningPlace)が保持するSnapの集合である" listedSnaps"を参照し、各SnapのJobオブジェクトを獲得する(ステップS111)。そして、獲得した各Jobのidから、それらと同じJobを指すSnapを検索する(ステップS112)。なお、この検索に用いる検索式は、例えばSQLなどを用いて、select\* from Snap where myJob=1 or myJob=2 or myJob=3、のように表される。

【0139】そして、検索結果を配列などで受け取り(ステップS113)、検索結果の中に登場する全ての

ユーザを抽出する(ステップS114)。なお、この検索結果は、例えば、snap[0]={1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, '(1,3,4)'}といったようなデータである。次いで、各ユーザごとにループを回し(ステップS115)、" createDate"が最新のSnapをピックアップして、これを各ユーザの最新Snapとする(ステップS116)。そして、上記のループ処理を全て終了したところで(ステップS117)、抽出した全てのSnapを関連状態保持部65に追加格納させ、これらをリスト表示部66に表示させる(ステップS118)。

【0140】次に、関連状態抽出部64が行う以前に行った仕事の検索処理を説明する。仕事を遂行している状況において以前に行った仕事の作業環境を抽出(再現)する必要が生ずる場合があり、図50にはその一例として、ユーザnomuraがJob1、Job2、Job3という3つのJobを現在状態に持っている場面で、以前行ったJobを検索するための指示が出された場合を示してある。この指示を出すことによって、自分が過去に作った全てのSnapから、Job1、Job2、Job3以外の全てのJobに関して最新のSnapが関連状態抽出部64によりデータベース61から検索される。そして、検索されたSnapが関連状態保持部65に保持され、リスト表示部66にそのリストが表示される。これにより、ユーザは過去に自分が他にどんな仕事をしてきたのかといった状態を一覧することができる。

【0141】この検索処理を図51に示すフローチャートを参照して更に詳しく説明する。まず、MyPastJobSnapボタン119ボタンが押されると(ステップS120)、自分のユーザid(この例では、ユーザnomuraのidの"1111")でデータベース61に格納されているSnapの検索が行われる(ステップS121)。この検索式としては、例えば、select \* from Snap where user=1111、といったようなSQL文が用いられる。そして、関連状態抽出部64はこの検索結果を受け取ると、検索結果の中に登場する全てのJobを抽出し(ステップS122)、さらに、現在開いているPlace(theOpeningPlace)が保持するSnapの集合である" listedSnaps"を参照して、各Snapが指すJobオブジェクトを獲得する(ステップS123)。

【0142】次いで、各Job毎にループを回し(ステップS124)、それらのSnapが現在開いているPlaceにあるSnapの指すジョブと同じか(つまり、Job1、2、3のどれかか)ということをチェックする(ステップS125)。この結果、同じであれば何もせずにループを繰り返す一方、Job1、2、3のいずれでもない場合には、各Job毎にSnapの生成日時(createDate)が最新のSnapを各Jobの最新スナップとする(ステップS126)。そして、上記



のループ処理を全て終了したところで(ステップS127)、抽出した全てのSnapを関連状態保持部65に追加格納させ、これらをリスト表示部66に表示させる(ステップS128)。

【0143】次に、現在状態導入部67が行う仕事の導入処理を説明する。仕事を遂行している状況において関連状態の現在状態への導入によるジョブの共有を行う必要が生ずる場合があり、本実施例では、Jobの共有(つまりコラボレーション)は、Snapを関連状態保持部65から現在状態保持部62へ導入することによって実現される。図52にはその一例として、関連状態保持部66に6つのSnapが抽出されている状況において、ユーザnomuraがこの中から「Job8(hayashi)9/18 14:23」を選択し、Import Snapボタン91を押して、このSnapを自分が現在開いているPlace(現在状態保持部62)に導入しようとしている場合を示してある。この導入を行うことによって、選択されたSnapがコピーされてユーザnomuraの開いているPlaceにJob8を指すSnapが保持されることになり、これにより、間接的にユーザhayashiの開いているPlaceのJob8とジョブ共有を行ったことになる。これにより、前述した関連作業環境抽出機能などを利用することが可能になる。

【0144】この導入処理を図53に示すフローチャートを参照して更に詳しく説明する。まず、Import Jobボタン91が押されると(ステップS130)、関連状態表示リスト101からSnapオブジェクトが1つ選択されているかどうかをチェックする(ステップS131)。この結果、選択されていない場合にはこのボタン押下による指示は無視される一方、選択されている場合には、そこで選択されているSnapオブジェクト(theSnap)を獲得する(ステップS132)。次いで、現在開いているPlace(theOpeningPlace)が保持するSnapの集合である“listedSnaps”を参照し(ステップS133)、theSnapが指すJobオブジェクト(theJob)が、listedSnapsの各Snapが指すJobに含まれているかをチェックする(ステップS134)。

【0145】この結果、含まれている場合には、同一のジョブを指しているSnapをPlaceのlistedSnapsから削除し、同じ順番に関連状態表示リストで選択されたSnapを挿入する(ステップS135)。そして、削除したSnapを関連状態保持部65に加え、そのスナップのisModified=Trueにする(ステップS136)。一方、theJobが含まれていない場合には、関連状態表示リストで選択されたSnapをスナップ集合の先頭に加え(ステップS137)、さらに、関連状態表示リストで選択されたSnapが指すジョブをジョブリストの先頭に加える(ステップS138)。

【0146】次に、上記した第3実施例における具体的

な利用例を説明する。なお、ここに示すプロトタイプは、Sun Microsystems社のJava言語によりプログラミングされている。ここでは、ユーザnomuraとユーザhayashiが共同作業をしている場合を例にとり、まず、図54には、“Nomura home”というPlaceにユーザnomuraがログインしている場合の画面表示例を示してある。ジョブリスト89には、RSP-Practices、Workstory、Evastory、Workstory-mk2、CooSS96、Information-Memo、PIG、test、CSCW、mass mediaという10個のJobが表示されている。これは、この“Nomura home”というPlaceが10個のSnapを現在状態保持部62に現在状態として保持しており、それらのSnapがジョブリスト89に表示されているJobをそれぞれ指し示していることを意味する。同図では、現在、RSP-PracticesというJobが選択されており、スナップビュー83には3つのMediatorが表示されている。

【0147】この状態から、Call Snapボタン76を押すと、図55に示すような画面表示がなされ、関連状態抽出のための検索指示を与えるウィンドウ(図44参照)が同一画面上に立ち上がって表示される。このウィンドウ中には、指示可能な全てのPlaceのリスト110、現在開いているPlaceのJobのリスト111、指示可能な全てのUserのリスト112が表示され、そして、Mediatorの名前を指示するフィールド113、Date(From)とDate(To)の期間を指定するフィールド114、115がある。なお、同図では、ユーザhayashiが、9/1の15:41から9/25の15:41までに生成したSnapを検索する指示を与えている。

【0148】そして、この状態から、Retrieveボタン116を押すと、図56に示すように検索指示ウィンドウ(図55)は閉じられ、蓄積部61に対して検索式が投げられる。そして、この検索結果が返ってくると、図56に示すように関連状態の表示リストを持つウィンドウ(図43参照)がポップアップして同一画面上に表示される。同図では、このリスト中の「<hayashi> Versioning 9/19[16:52]」を選択して、そのSnapが保持するMediatorをスナップビュー83に表示している。

【0149】そして、ユーザhayashiの行っている「Versioning」という仕事に関し、共同作業を始めようとした場合、Import Snapボタン91を押すことにより、関連状態表示リスト内で選択されていたSnapを現在状態、つまり今開いているPlaceに加えることができる。これを実行した後の画面表示が図57に示してあり、こうすることによって、ユーザnomuraは「Versioning」という仕事をユーザhayashiとともに進める事が可能になる。この後、ユーザnomuraは随時What's New機能を使って、ユーザhayashiが「Versioning」という仕事に関してどんな新たな情報を追加したのかを知ることができる。さらに、ある程度2人の仕事が進んだと

ところで、Gather Job機能、つまり2人の作業環境の和集合を後述するように状態合成部70の処理によってとることにより、共通の仕事の場を作り、最終的なアウトプットのまとめを行うことができる。

【0150】図58には、本発明の第4実施例に係る情報処理装置を示してある。なお、上記した第3実施例を同一部分には同一符号を付して重複する説明は割愛する。本実施例の情報処理装置は、第3実施例の構成に、現在状態保持部62が保持する2以上の情報単位集合中の情報単位に対して集合演算を施す状態合成部70が加えられたものである。状態合成部70は、現在状態保持部62が保持している複数のSnapに対して、これらSnapに含まれているMediatorを集合演算し、この結果を現在状態保持部62に保持させる処理を行う機能手段である。

【0151】また、状態合成部70は、表示画面(図42参照)のCombineボタン78が押されると、図59に示すような状態合成指示画面ウィンドウを表示させ、このウィンドウにおいて複数のSnapを合成させる指示がユーザから入力されると、所定の集合演算処理を実行する。状態合成指示画面ウィンドウには、各種の情報を表示するリスト部、各種情報を入力するためのフィールド、及び、各種の機能を実行させるためのボタンが設けられており、これら各部の機能は次のようになっている。現在状態リスト部120は現在状態保持部62が保持する全てのSnapの指すJob名のリストを表示する領域であり、関連状態リスト部121は関連状態保持部65が保持する全てのSnapの属性(選択情報)のリストを表示する領域である。

【0152】base snapボタン122は、2つのSnap間で差分を演算する場合等において、一方のSnapをベースとして保持させるボタンである。すなわち、通常はsingle selectionであるので、1つのSnapを選んだ後に、他のSnapを選ぶと選択状態が移ってしまう。そこで、このボタン122を1つのSnapを選んだ後に押すと、その選ばれたSnapをベースSnapとし、次からmultiple selection可能に切り替える機能が実行され、ユーザはこのボタン122を押した後は、続けて複数のSnapを選択状態にすることができる。このようにベースSnapとされたSnapは、差分をとるときに引かれる側のSnapとなる。なお、NewJobNameフィールド127に何も文字列が書かれていないときに、or (+) ボタン123、subtraction (－) ボタン124、and (\*) ボタン125のいずれかが押されると、ベースSnapのJob名で状態合成部70による集合演算結果のSnapが現在状態保持部62に挿入格納される。

【0153】or (+) ボタン123は、現在状態リストと関連状態リストとで複数選択された全てのSnapが指すMediator集合に関して、状態合成部70

に和集合演算を行うことを指示するボタンである。なお、この和集合をとる際に同じ内容のMediatorは重複して要素としないことが行われ、この「同じ」を判断するために、本実施例では、Mediatorの名前とURLの一致を検査している。但し、この検査方法に限る必要はなく、例えばメモや日時までチェックしたり、他のオブジェクトとの関連までチェックして同一性を判断したり、リファレンス先の文書のバージョンの同一性までチェックしてもよい。

【0154】subtraction (－) ボタン124は、ベースSnapから、他の選択された複数のSnapの差分演算を指示するボタンである。つまり、状態合成部70にsubtractionSnap=BaseSnap-(Snap1+Snap2+...)という演算を行わせる。and (\*) ボタン125は、現在状態リストと関連状態リストとで複数選択された全てのSnapが指すMediator集合に関して、状態合成部70に共通演算を行うことを指示するボタンである。Closeボタン126は、当該状態合成指示画面を閉じさせる指示を行うボタンである。

【0155】NewJobNameフィールド127は、集合演算結果を現在状態保持部62に挿入保持させる際に、その演算結果であるSnapのJobを当該フィールド127に書かれた文字列のジョブ名にさせるものであり、このフィールド127が空である場合には演算結果はベースSnapのジョブ名になる。Gather Jobボタン128はジョブ成果集約機能と呼び出すボタンであり、この詳細は後述する。What's Newボタン129はWhat's New機能と呼び出すボタンであり、この詳細は後述する。

【0156】Job Essenceボタン130はJob Essence機能と呼び出すボタンであり、この機能は、複数人によりジョブを共有して仕事を進めている場合に、そのジョブに関して各人の最新のSnapの共通集合をとって表示させるものである。これにより、共有しているミッション、参考文献、提出先など、必要不可欠の情報をきちんとシェアしているかを確認することができる。また、新たにジョブに加わるメンバが、共有情報から場を導出して環境を整えることができる。

【0157】次に、本実施例に係る情報処理装置の集合演算処理を説明する。まず、和集合演算(ジョブ成果集約機能)について、図60に示すジョブの成果集約のための状態合成例を参照して説明する。この例において、ユーザnomuraがJob1、Job2、Job3の3つのJobを現在状態に持っている場面で、ジョブ成果集約指示を出したとすると、Job1、Job2、Job3の各Jobに関して他ユーザの最新のSnapがデータベース61から検索される。この例では、Job1に関してsekijima、hayashi、hazamaの3人の最新Snapが、Job2に関してsekijima、hayashiの2人の最新Snapが検索されている。そして、各Job毎にnomu

raのSnapも含めて、検索されたSnapの和集合演算が行われ、その結果が例えばJob1(+)、Job2(+ )というジョブ名で現在状態保持部62に挿入される。

【0158】このジョブ成果集約機能(和集合演算処理)を図61に示すフローチャートを参照して更に詳しく説明する。まず、GatherJobボタン128が押されると(ステップS140)、現在開いているPlace(theOpeningPlace)が保持するSnapの集合であるlistedSnapsを参照し、各SnapのJobオブジェクトを獲得する(ステップS141)。そして、獲得した各Jobのidから、それらと同じJobを指すSnapを、例えば、select \* from Snap where myJob=1 or myJob=2 or myJob=3、というSQL文で検索する(ステップS142)。

【0159】そして、この検索結果を、例えば、snap[0]=[1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, '{1,3,4}']、といった配列で受け取り、(ステップS143)、この検索結果の中に登場する全てのUserを抽出する(ステップS144)。そして、各Userごとにループを回し(ステップS145)、createdDateが最新のSnapを各ユーザーの最新Snapとする(ステップS146)。次いで、各ユーザーの最新Snapの数だけループを回し(ステップS147)、さらに各Snapが保持するMediatorの数だけループを回す(ステップS148)。その中でさらに、合成Snapに既に加えられた全てのMediatorの数だけループを回し(ステップS149)、既に加えられたMediatorと新たなMediatorが同じかどうかをチェックし(ステップS150)、同じでない場合には新たなMediatorを合成Snapに加える(ステップS151)。このような処理を、ループするSnapとMediatorとがなくなるまで繰り返す(ステップS152)。

【0160】次に、差分演算処理(What's New機能)について、図62に示すように自分の行っているJobに関する新しい情報を抽出する(すなわちWhat's New機能)ための状態合成例を参照して説明する。この例において、例えばユーザーnomuraがJob1、Job2、Job3の3つのJobを現在状態に持っている場面、What's Newボタン129が押されてWhat's New命令が出されたとする。なお、同図では、そのユーザーnomuraが保持するSnapはA1、B1、C1として示されている。この命令の結果、Job1、Job2、Job3の各Jobに関して他のユーザーの最新のSnapで、ユーザーnomuraのSnapよりも新しいものがデータベース61から抽出される。つまり、Job1に関してはA1よりも新しい、ユーザーsekijimaのA2、B2、ユーザーhayashiのA4が抽出される。

【0161】ここで、What's New機能により抽出したい

情報は、例えばJob1に関してはユーザーnomuraがA1を作成以降に他のユーザーが生成した情報なので、新規Snapとして抽出されたユーザーsekijimaのA2、B2、ユーザーhayashiのA4に関して、ユーザーnomuraのSnapの直前に作られた、そのユーザーのそのJobのSnapを抽出(図62ではA3、B4、A5)し、これらとの差分をとることで、ユーザーnomuraが見ていない新しい情報が何であるかを知ることができる。つまり、新規情報=Σ(ユーザーXの新規Snap-ユーザーXの直前Snap)-nomuraSnap、という計算式(ただしここでは、“+”は和集合合成演算、“-”は差分合成演算、Σは比較対象となるユーザー全員に対して行った結果の和集合をとることを示す。)により、ユーザーnomuraの新規情報を得ることができる。なお、図62の例では、Job1(new)=(A2-A3)+(A4-A5)-A1、Job2(new)=(B2-B4)-B1、といった計算が行われることになる。

【0162】このWhat's New機能(差分集合演算処理)を図63に示すフローチャートを参照して更に詳しく説明する。まず、What's Newボタン129が押されると(ステップS160)、現在開いているPlace(theOpeningPlace)が保持するSnapの集合であるlistedSnapsを参照して、この中から幾つかの元Snapを獲得し(ステップS161)、それら元Snapの生成日時を獲得する(ステップS162)。次いで、獲得した各Jobのidから、それらと同じJobを指すSnapを、例えば、select \* from Snap where myJob=1 or myJob=2 or myJob=3、といったSQL文で検索する(ステップS163)。

【0163】そして、この検索結果を例えば、snap[0]=[1, 'Wed Jul 10 15:00:00 JST 1996', 1111, 1, 0, '{1,3,4}']といった配列で受け取り(ステップS164)、この検索結果の中に登場する全てのUserを抽出し(ステップS165)、各Userごとにループを回し(ステップS166)、検索結果の各SnapのcreatedDateが、同じJobを指す元SnapのcreatedDateよりも新しく、且つ、そのユーザーのそのJobで最新のSnap(図62のA2、B2、A4)を、各ユーザーの最新スナップとして抽出する(ステップS167)。次いで、抽出された各最新Snapと同じユーザー又は同じジョブで、且つ、同じJobを指す元SnapのcreatedDateの以前に作られたSnapで、最新のもの(図62のA3、B4、A5)を、各ユーザーの直前Snapとして抽出する(ステップS168)。

【0164】そして、各Job毎に最新Snapと直前Snapの差分をとり、その和集合をとる(ステップS169)。さらに、各Jobの元Snapから和集合Snapの差分をとり、それをそのJobの新参Mediator集合として、「ジョブ名(new)」という名前のJobを作る(ステップS170)。次いで、和集合SnapのJob(つまり「ジョブ名(new)」)がlistedS

napの各Snapが指すJobに既に含まれているかを判断し(ステップS171)、含まれている場合には、同一のJobを指しているSnapをlistedSnapsから削除し、同じ順番に関連状態表示リストで選択されたSnapを挿入する(ステップS172)。さらに、削除したSnapを関連状態保持部65に加え、そのSnapについてisModified=Trueにする(ステップS173)。一方、含まれていない場合には、和集合SnapをSnap集合の先頭に加え(ステップS174)、和集合Snapの指すJobをJobリストの先頭に加える(ステップS175)。このような処理を、ループするUserがなくなるまで繰り返し行う(ステップS176)。

【0165】次に、上記した第4実施例における具体的な利用例を説明する。なお、ここに示すプロトタイプは、Sun Microsystems社のJava言語によりプログラミングされている。ここでは、ユーザnomuraとユーザhayashiが共同作業をしている場合を例にとり、まず、図64には、ユーザnomuraが「Nomura home」というPlace63を開いており、さらに、関連状態表示リストのウィンドウ140と、状態合成指示ウィンドウ(図59参照)を開いている状態を示してある。なお、スナップビュー83には関連状態表示リスト中で選択されている「<hayashi>P1G 9/10[15:51]」が表示されている。

【0166】この状態で、自分が行っている仕事である「CooSS96」とユーザhayashiの過去の場の状態である「<hayashi>P1G 9/10[15:51]」を合わせて、新たな仕事を始めようとしたとする。この場合には、まず、状態合成指示ウィンドウ中の左のリスト120(つまり現在状態リスト)中の「CooSS96」を選びbase snapボタン122を押し、続いて、右のリスト121(つまり関連状態リスト)から「<hayashi>P1G 9/10[15:51]」を選び、さらに、New Job Nameフィールド127に「New Project」と打ち込んだ後、Gather Jobボタン128を押す。

【0167】この結果、現在状態が保持する「CooSS96」というJobを指しているSnapと、関連状態が保持する「<hayashi>P1G 9/10[15:51]」のSnapの和集合演算が行われ、結果として、図65に示すように「New Project」という新しいJobが生成されて現在状態保持部62に追加格納され、現在状態(つまり今開いているPlace)に挿入される。また、スナップビュー83には、2つのSnapが保持していたMediatorが全て表示されている。但しこのとき、和集合を行うSnapに同じMediatorが含まれていた場合は、そのMediatorは重複してコピーされない。

【0168】

【発明の効果】以上の説明から明らかなように、本発明の情報処理装置によれば、ある業務に必要な情報を情報

単位保持手段により保持し、その作業環境を用いて業務を遂行することができるとともに、選択情報を指示することによって、情報単位集合保持履歴記憶手段が記憶する所望の情報単位集合を呼び出し、仕事の場の状態を切り替え、そこから別の仕事を再開することができる。また、前もって分類を行わなくても所望の作業環境が得られ、仕事が進んだ後でも、仕事の名前、日時、利用者名、情報単位名など様々な視点で作業環境を呼び出すことができる。また、過去の仕事の状態を保持しているため、過去の時点の状態に戻って仕事をスタートさせることや、過去の状態を復元することにより、そこからこれからの仕事に必要な情報を抜き出してくることができる。

【0169】また、選択情報付与手段によって情報単位集合に日時・利用者名・仕事名などの選択情報を付与して情報単位集合保持履歴記憶手段に記憶し、情報単位集合保持履歴分類表示手段に情報単位集合を選択情報によって分類表示するようにしたため、作業環境の切り替えを利用者に理解しやすく提示することができる。また、情報単位集合に複数の選択情報を付与しておき、情報単位集合保持履歴分類表示手段における分類項目を利用者の要求に合わせて変更することにより、業務を多角的に進めることができる。また、1つの仕事を行なっている仕事の場の状態を複数保存しておくことにより、過去の仕事の状態や関連情報を参照したり、過去の仕事の状態に戻って新たな仕事を始めることができる。

【0170】また、仕事の場の状態の履歴が蓄積されるため、ある仕事に必要な情報は何か、ある仕事でよく使われる情報は何か、或いはある仕事に固有の情報は何かといった、仕事のノウハウを抽出・再利用することができる。また、過去の仕事の場の状態から、必要な情報を抽出することにより、利用者は新たな環境を構築することができる。また、いくつかの過去の仕事の場の状態を合成することによって、利用者は新たな環境を構築することができる。

【0171】更に具体的には、自分のみならず他人の過去から現在にかけての活動経過や成果を活用して新たな仕事に必要な環境を生成し、そこで仕事を始めることが可能となり、効率的な仕事を実現することができる。また、緩い協調メンバー間(コミュニティ)においても、お互いの知識の交換を行い易い枠組みを提供することができる。また、グループでの情報共有や進捗管理においては、従来のような単一業務におけるコラボレーションではなく、他人の仕事のアクション、状況、進捗を反映できる総合的なコラボレーションを可能とすることができる。また、現在行っているジョブ以外は現在状態から削除してしまってもこれを復元することが容易な枠組みを提供しているため、ユーザによる仕事の能率が飛躍的に向上する。

【図面の簡単な説明】

【図1】 本発明の第1実施例に係る情報処理装置の構成図である。

【図2】 情報単位集合と選択情報との関係を説明する概念図である。

【図3】 情報単位集合の情報単位集合保持履歴記憶手段への記憶と読出を説明する概念図である。

【図4】 仕事の種類の表明と仕事の切替を説明する概念図である。

【図5】 情報単位集合と選択情報リストの表示の一例を示す図である。

【図6】 情報単位集合と選択情報リストの表示の他の一例を示す図である。

【図7】 情報単位集合と選択情報リストの表示の更に他の一例を示す図である。

【図8】 利用者に関するデータ構造の一例を示す説明図である。

【図9】 ワークスペースに関するデータ構造の一例を示す説明図である。

【図10】 ジョブに関するデータ構造の一例を示す説明図である。

【図11】 ワークスペース-ジョブ対応のデータ構造の一例を示す説明図である。

【図12】 情報単位集合に関するデータ構造の一例を示す説明図である。

【図13】 情報単位に関するデータ構造の一例を示す説明図である。

【図14】 ワークスペース-情報単位集合対応のデータ構造の一例を示す説明図である。

【図15】 情報単位集合-情報単位対応のデータ構造の一例を示す説明図である。

【図16】 選択情報に関するデータ構造の一例を示す説明図である。

【図17】 選択情報リストに関するデータ構造の一例を示す説明図である。

【図18】 蓄積されたコマンド列の一例を示す説明図である。

【図19】 記憶実行ルールの一列を示す説明図である。

【図20】 情報単位取り込み処理の手順の一例を示すフローチャートである。

【図21】 情報単位集合の記憶判断処理の手順の一例を示すフローチャートである。

【図22】 新規ジョブ登録処理の手順の一例を示すフローチャートである。

【図23】 選択情報の分類表示処理の手順の一例を示すフローチャートである。

【図24】 情報単位集合の読み込み処理の手順の一例を示すフローチャートである。

【図25】 情報単位集合と選択情報リストの表示の一例を示す図である。

【図26】 情報単位集合と選択情報リストの表示の一例を示す図である。

【図27】 情報単位集合と選択情報リストの表示の一例を示す図である。

【図28】 情報単位集合と選択情報リストの表示の一例を示す図である。

【図29】 情報単位集合と選択情報リストの表示の一例を示す図である。

【図30】 本発明の第2実施例に係る情報処理装置の構成図である。

【図31】 複数の情報単位集合の合成指示を説明する図である。

【図32】 合成する情報単位集合の選択指示を説明する図である。

【図33】 フィルタリング指示を説明する図である。

【図34】 複数の情報単位集合の合成処理を説明する概念図である。

【図35】 和集合演算処理の手順の一例を示すフローチャートである。

【図36】 共通集合演算処理の手順の一例を示すフローチャートである。

【図37】 差分演算処理の手順の一例を示すフローチャートである。

【図38】 フィルタリング処理の手順の一例を示すフローチャートである。

【図39】 本発明の第3実施例に係る情報処理装置の構成図である。

【図40】 データベースのスキーマを説明する図である。

【図41】 データ構造を説明する図である。

【図42】 メイン画面の表示例を示す図である。

【図43】 関連状態として保持されたスナップの表示リスト画面の一例を示す図である。

【図44】 関連状態抽出のための検索指示画面の一例を示す図である。

【図45】 作業環境保存のタイミングを説明する図である。

【図46】 保存タイミング処理の手順を示すフローチャートである。

【図47】 データベースへの格納処理の手順を示すフローチャートである。

【図48】 自分と関連のある仕事の作業環境抽出のための検索処理を説明する図である。

【図49】 関連環境抽出処理の手順を示すフローチャートである。

【図50】 自分が以前行った仕事の作業環境抽出のための検索処理を説明する図である。

【図51】 自分の以前の仕事の環境抽出処理の手順を示すフローチャートである。

【図52】 関連状態の現在状態への導入処理を説明す

る図である。

【図53】 導入によるジョブ共有処理の手順を示すフローチャートである。

【図54】 メイン画面の表示例を示す図である。

【図55】 既に保存したSnapの検索指示画面を一例を示す図である。

【図56】 検索結果の1つを指示してスナップをスナップビューに表示させた画面表示例を示す図である。

【図57】 スナップを現在状態保持部に導入した場合に画面表示例を示す図である。

【図58】 本発明の第4実施例に係る情報処理装置の構成図である。

【図59】 状態合成指示の画面の一例を示す図である。

【図60】 ジョブの成果集約のための状態合成を説明する図である。

【図61】 ジョブ成果集約機能の処理手順を示すフローチャートである。

【図62】 自分の行っているジョブに関する新しい情報を抽出する状態合成を説明する図である。

【図63】 What's New機能の処理手順を示すフローチャートである。

ートである。

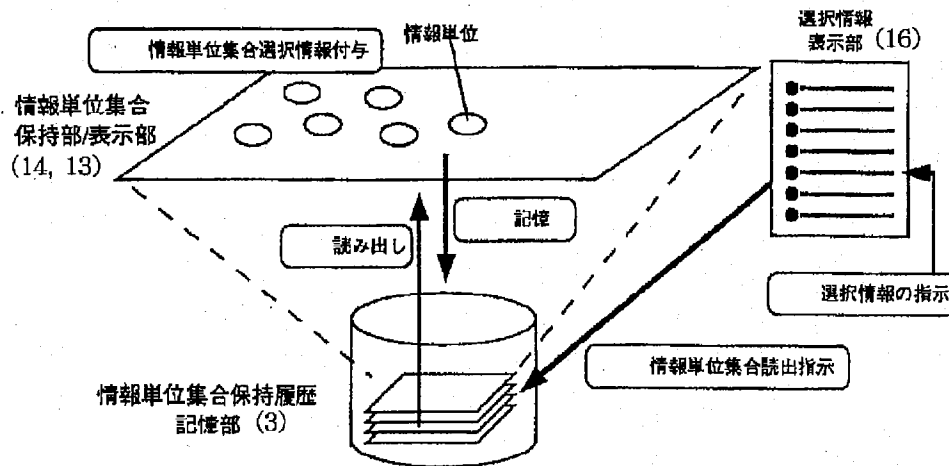
【図64】 2つのスナップの合成指示を入力する画面表示を示す図である。

【図65】 2つのスナップの合成演算結果を表示した画面を示す図である。

【符号の説明】

1・・・サーバ、 2・・・情報単位記憶部、 3・・・情報単位集合保持履歴記憶部、 4・・・情報単位集合選択部、 5・・・情報単位集合読出部、 6・・・選択情報付与部、 7・・・記憶指示部、 8・・・操作コマンド蓄積部、 9・・・情報単位集合記憶判断部、 10・・・ネットワーク、 11・・・クライアント、 12・・・入出力部、 13・・・情報単位集合表示部、 14・・・情報単位集合保持部、 15・・・情報単位編集部、 16・・・選択情報リスト表示部、 17・・・選択情報指示部、 61・・・蓄積部（データベース）、 62・・・現在状態保持部、 63・・・現在状態表示部、 64・・・関連状態抽出部、 65・・・関連状態保持部、 66・・・リスト表示部、 67・・・関連状態導入部、 68・・・保存判断部、 69・・・現在状態獲得部、 70・・・状態合成部、

【図3】



【図8】

ユーザId	user2
ユーザ名	AokI
パスワード	*****

【図10】

ジョブId	job1
ジョブ名	特許作成スペース集合
作成日時	1995.5.12,14:12
ユーザ	user2

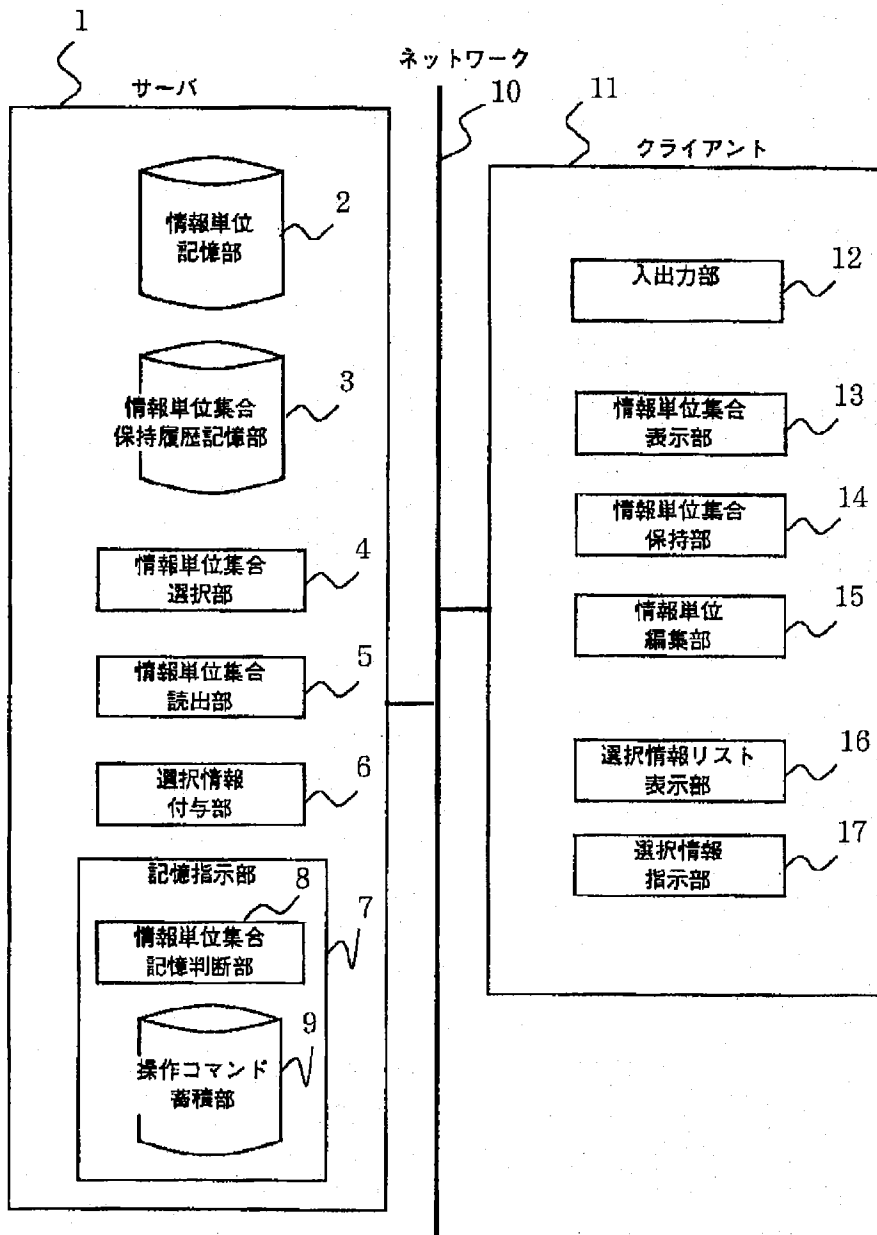
【図9】

ワークスペースId	ws1
ワークスペース名	青木ホームスペース
カレント情報単位集合	set1
選択情報リスト	fst1

【図11】

ワークスペースid	ws1
ジョブid	job1

【図1】



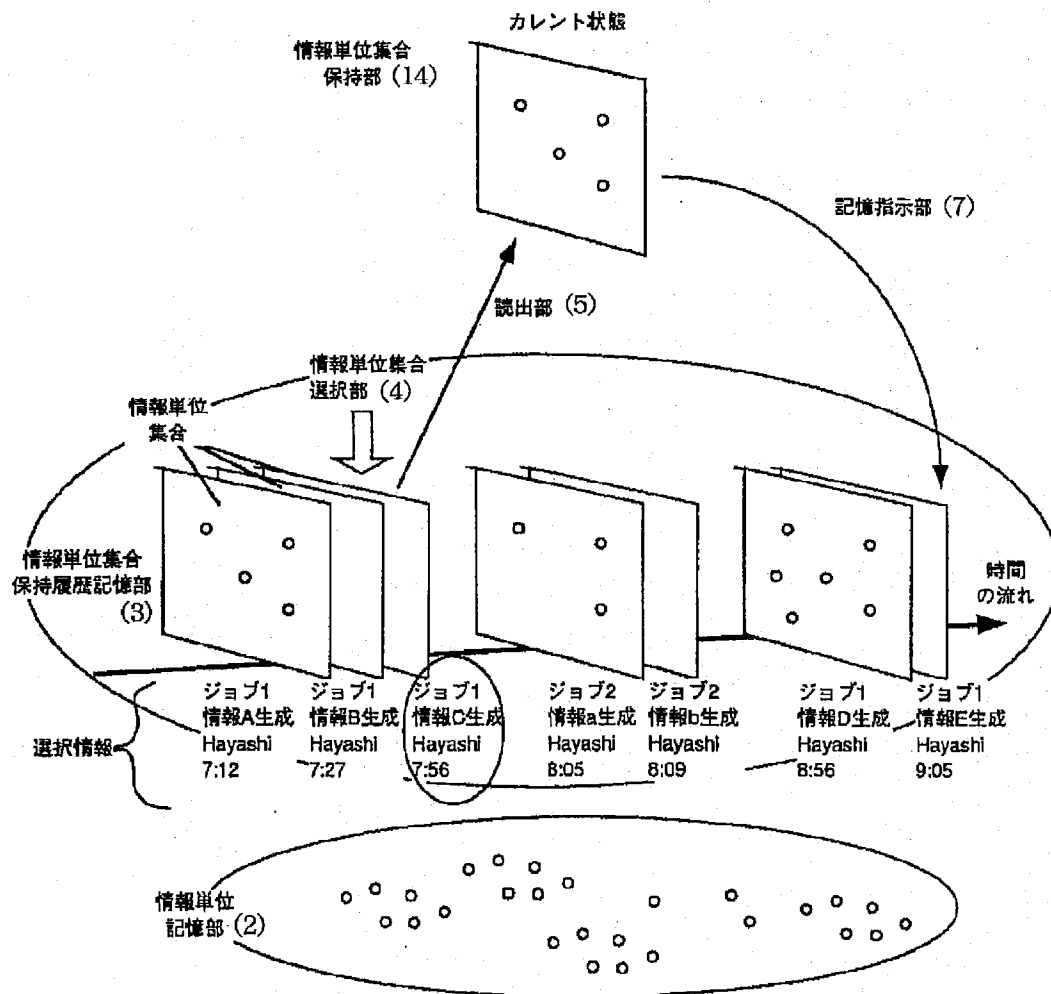
【図12】

情報単位集合id	set 1
所属するワークスペース	ws 1
作成日時	1995.5.12,14:12
ユーザ	user 2
ジョブ	job 1
流入情報単位	unit 8

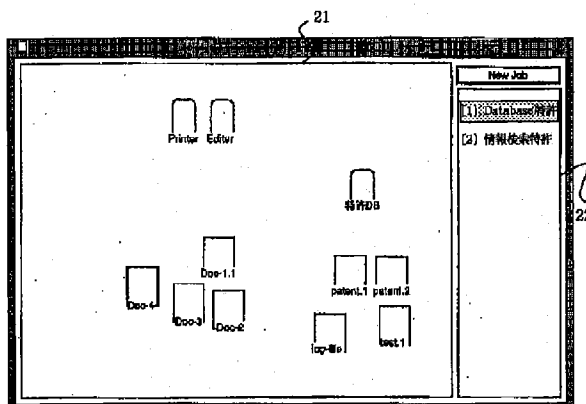
【図13】

情報単位id	unit 5
情報単位名	特許フォーマット
日時	1995.5.12,15:20
ユーザ	user 2
参照オブジェクト	~/doc/Pat-form
起動アプリケーション	~/bin/edit
コメント	重要度=高い
配属	{34,56}

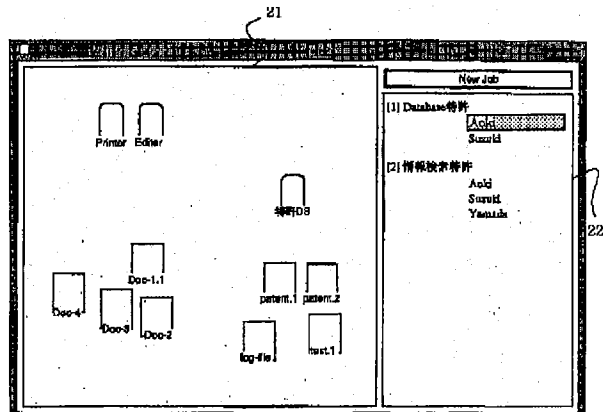
【図2】



【図5】

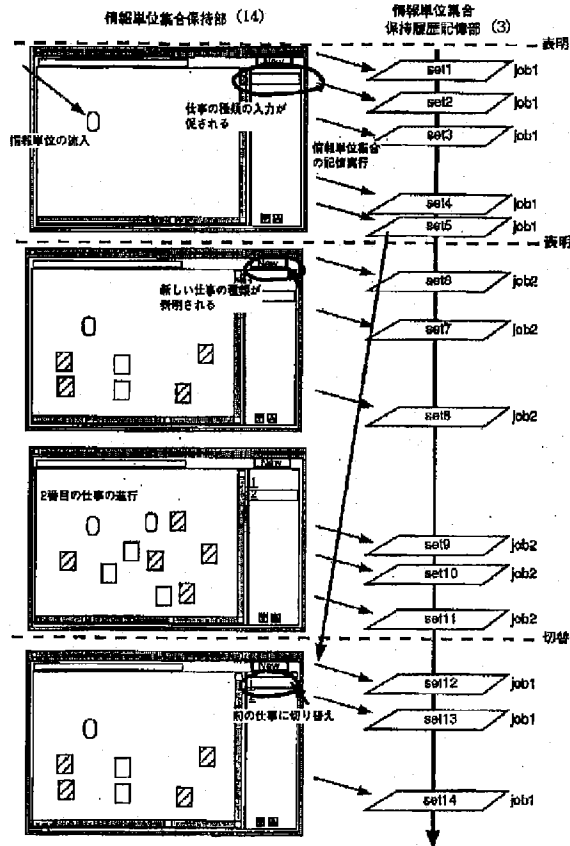


【図6】





【図4】



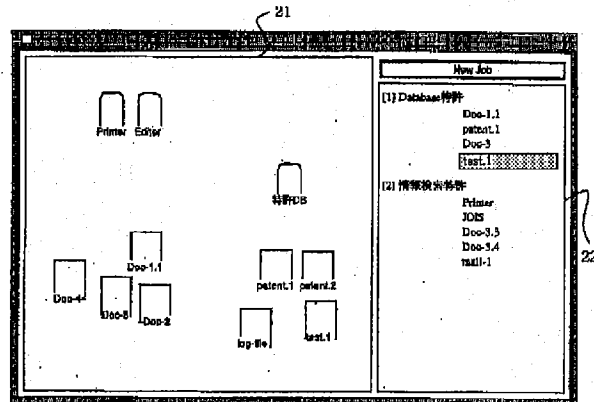
【図14】

ワークスペースid	ws1	情報単位集合id	set1
情報単位集合id	unit1	情報単位id	unit1

【図17】

選択情報リストid	list1
第1層選択情報id	select1
第2層選択情報id	select3
作成日時	1995.5.12,14:12
ユーザ	USER2

【図7】



【図16】

選択情報id	選択情報属性
select1	ジョブ
select2	作成者
select3	流入情報単位
select4	作成日時

【図18】

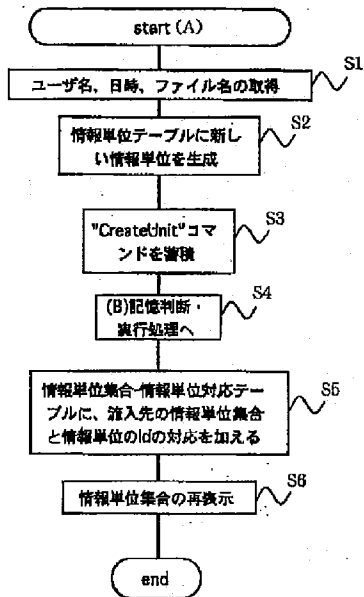
【図15】

No.	操作コマンド
1	OpenWS
2	CloseWS
3	OpenWS
4	SelectJob
5	SwitchSet
6	DeleteUnit
7	CreateUnit
8	CreateJob
9	RemoveAllUnit
:	:

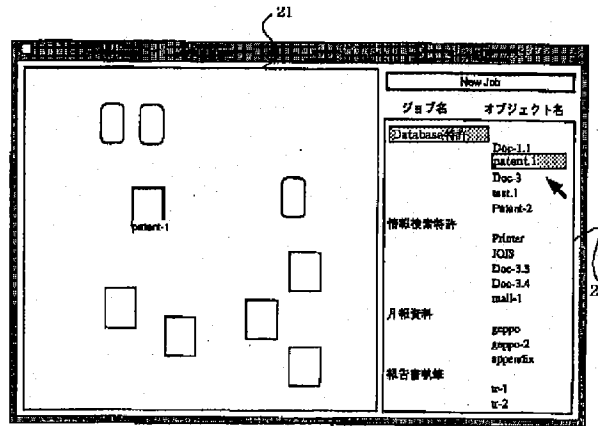
【図19】

- 新規作成されたワークスペースに、新たに情報単位が生成されたとき  
stack(n-1)="CreateWS" and stack(n)="CreateUnit";
- 情報単位を全て消去してから新たに情報単位が生成されたとき  
stack(n-1)="RemoveAllUnit" and stack(n)="CreateUnit";
- ジョブを切り替えた後、情報単位が生成されたとき  
stack(n-2)="SelectJob" and stack(n-1)="SwitchSet" and stack(n)="CreateUnit";

【図20】

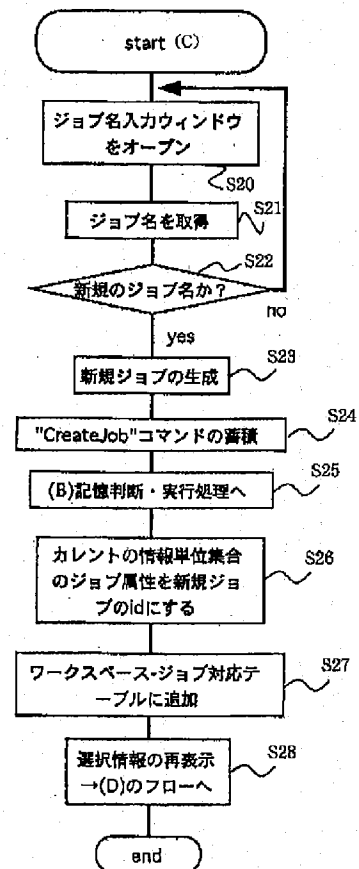
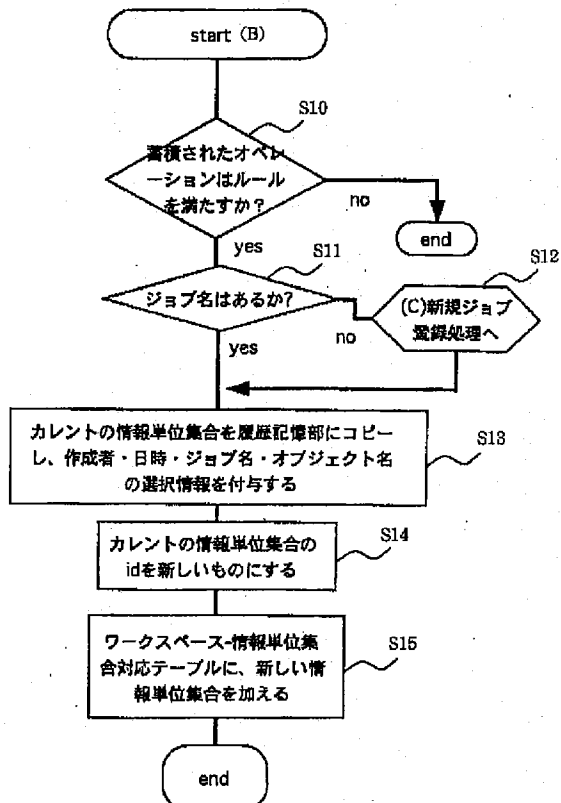


【図27】

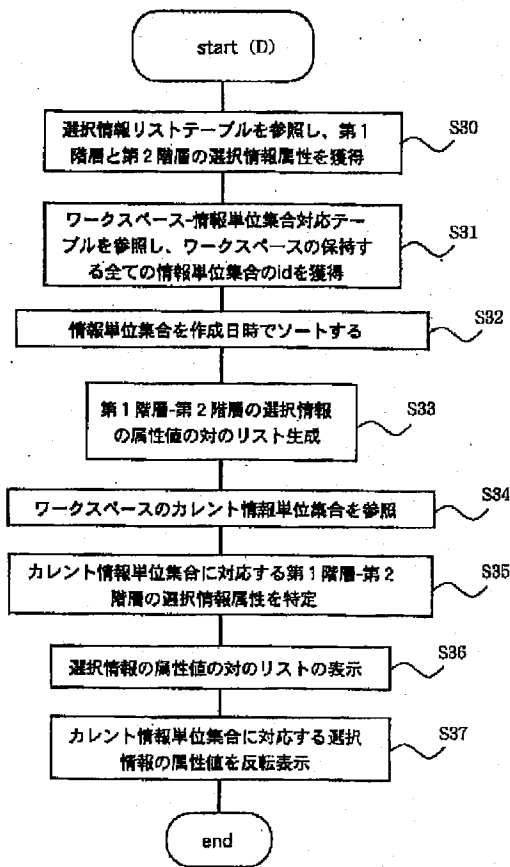


【図22】

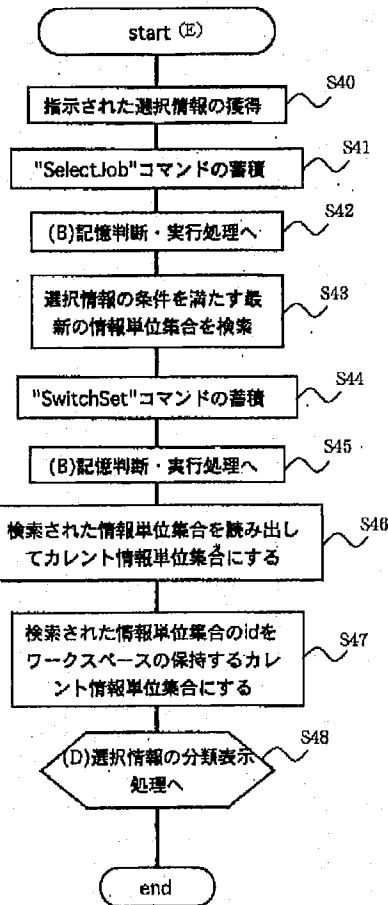
【図21】



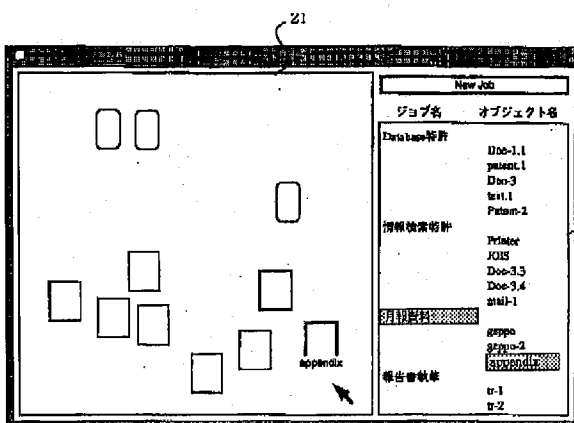
【図23】



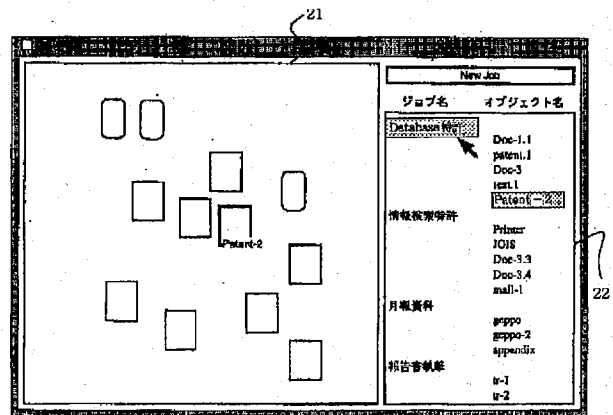
【図24】



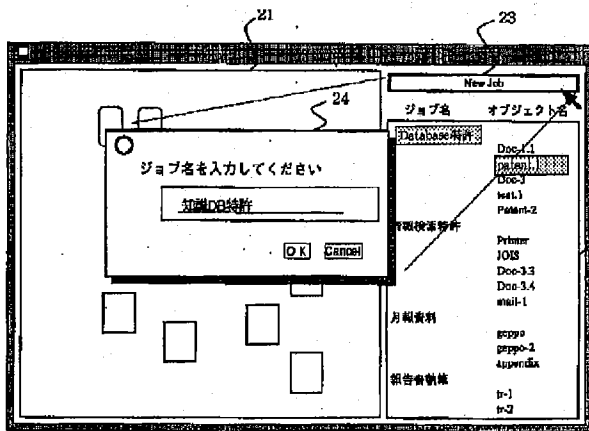
【図25】



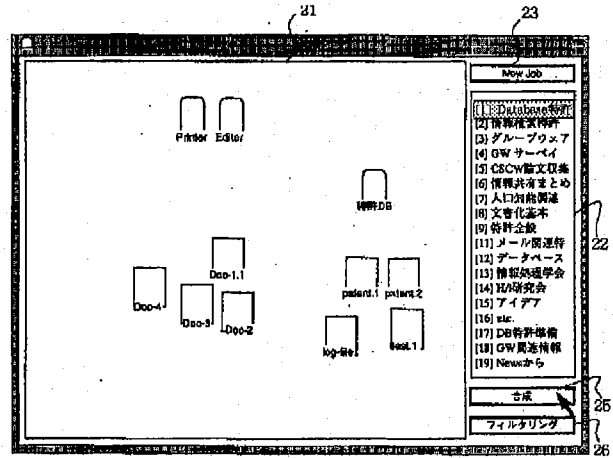
【図26】



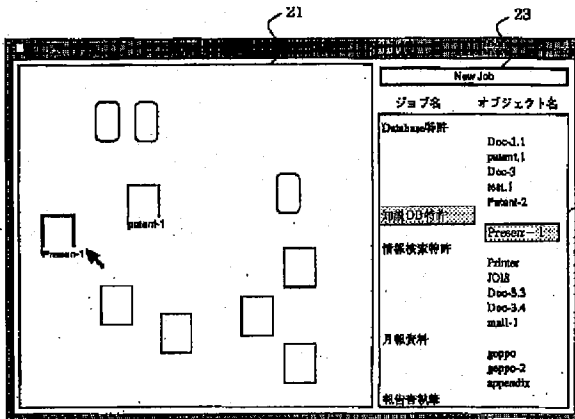
【図28】



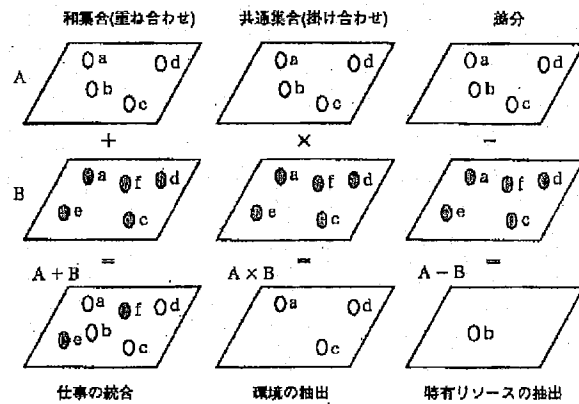
【図31】



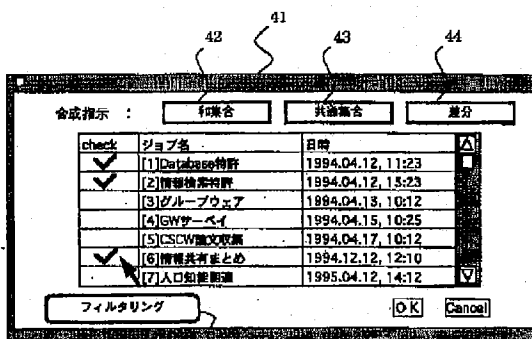
【図29】



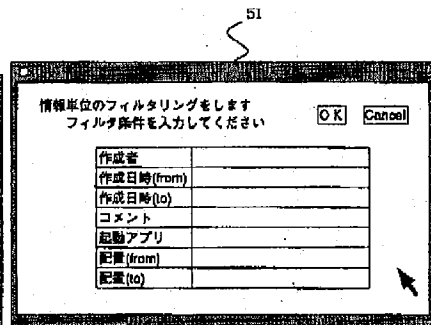
【図34】



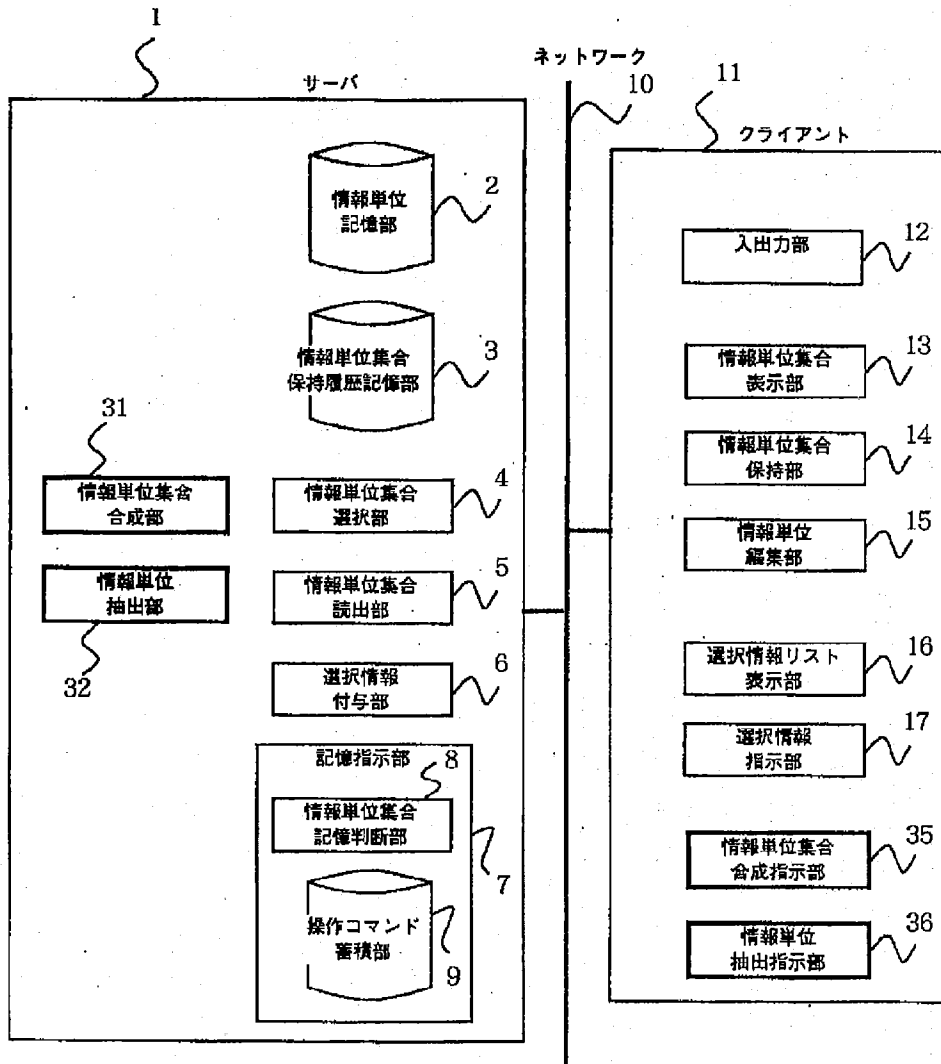
【図32】



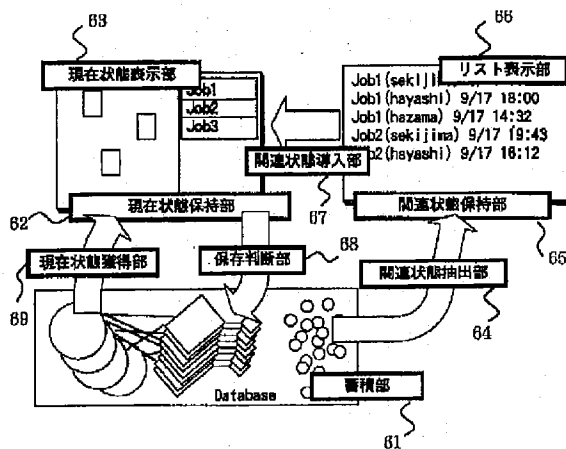
【図33】



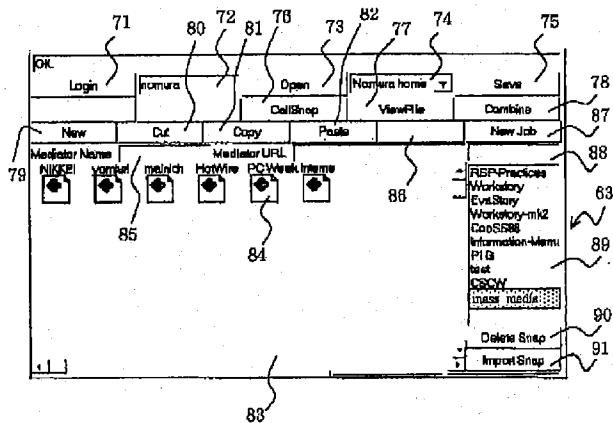
【図30】



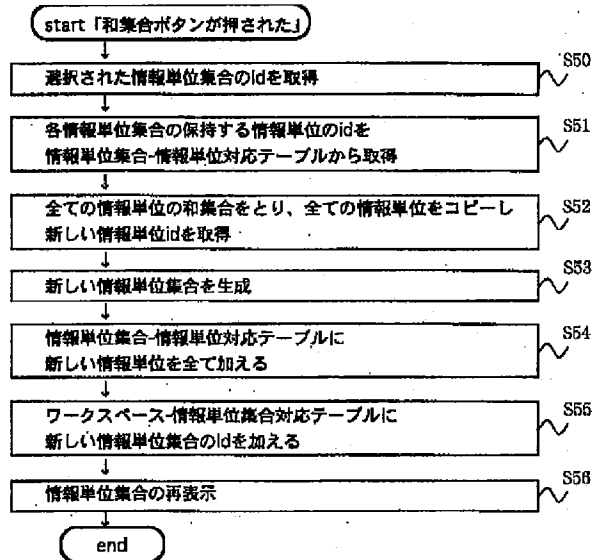
【図39】



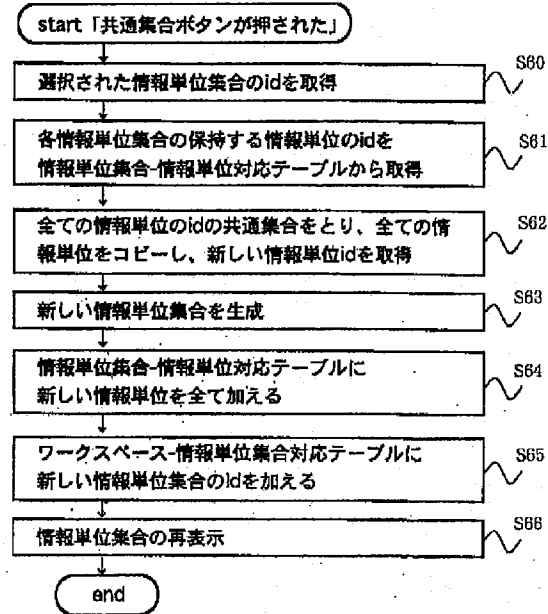
【図42】



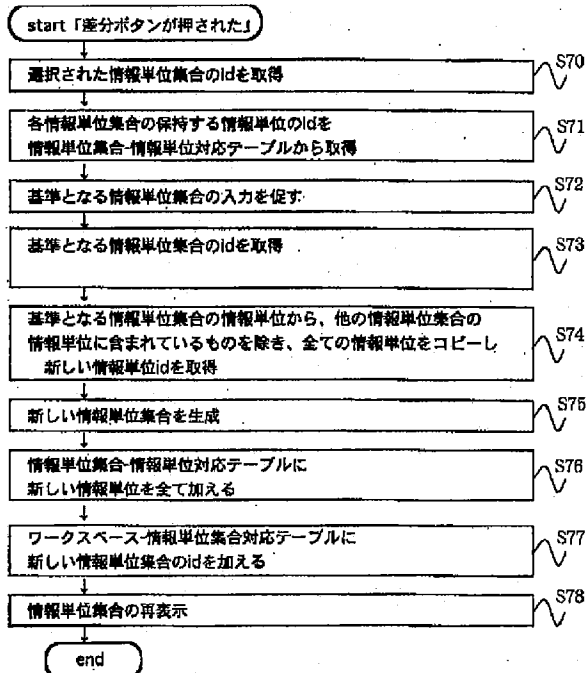
【図35】



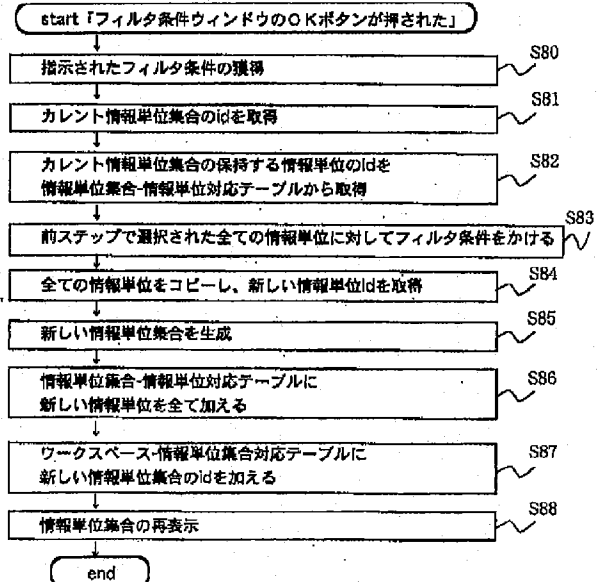
【図36】



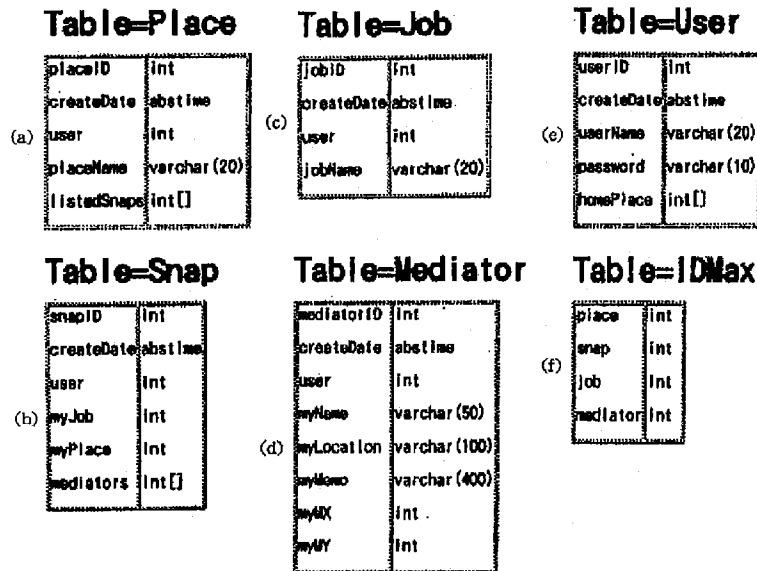
【図37】



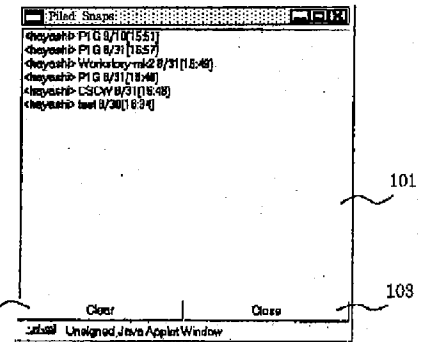
【図38】



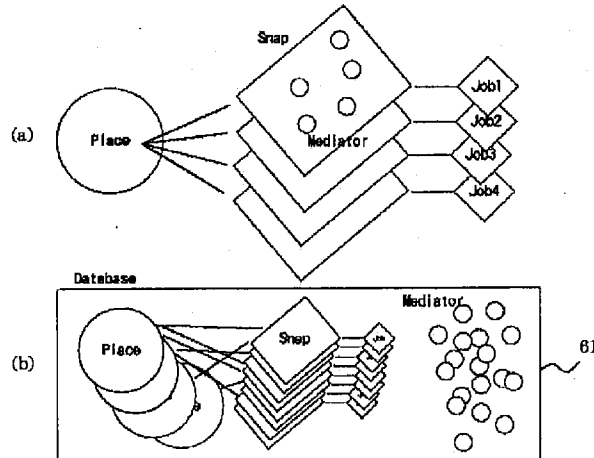
【図40】



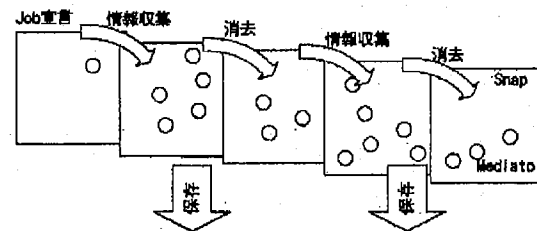
【図43】



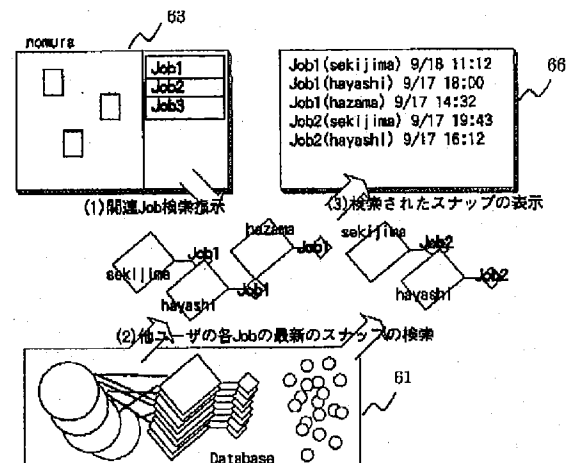
【図41】



【図45】



【図48】



【図44】

You can get the snaps in the database on condition that...

Place	Job	User	Mediator name
Nomura home	RSP-Practices	nomura	
Hayashi home	Workstory	hayashi	
Sekijima home	EvsStory	kunimoto	
Hazama home	Workstory-mk2	hashimoto	
Suzuki home	CoosS96	takemoto	
Takemoto home	Information-Memo	suzuki	
Hashimoto home	PiG	hazama	
Kunimoto home	test	sekijima	
Adnuma home	CSCW	guest	
Ichimura home	mass media	yasumatsu	
Katsurabayashi home		masada	
Kamibayashi home		fujimoto	
Kuma home		shaya	
Koda home		sasaki	
Sasaki home		koda	

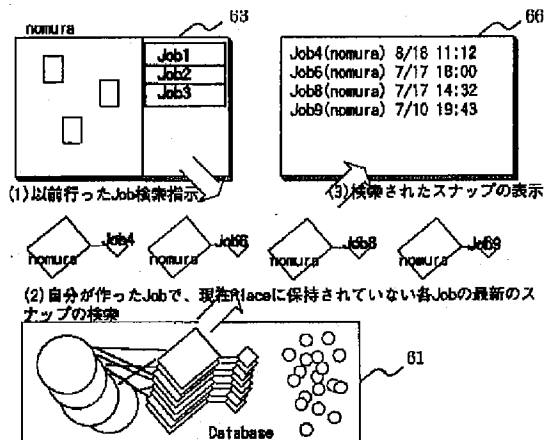
Retrive Close

Shared Job Snaps My Past Job Snaps

Unsigned Java Applet Window

【図50】

【図57】



OK Login nomura Open Nomura home Save

CellSnap ViewFile Combine

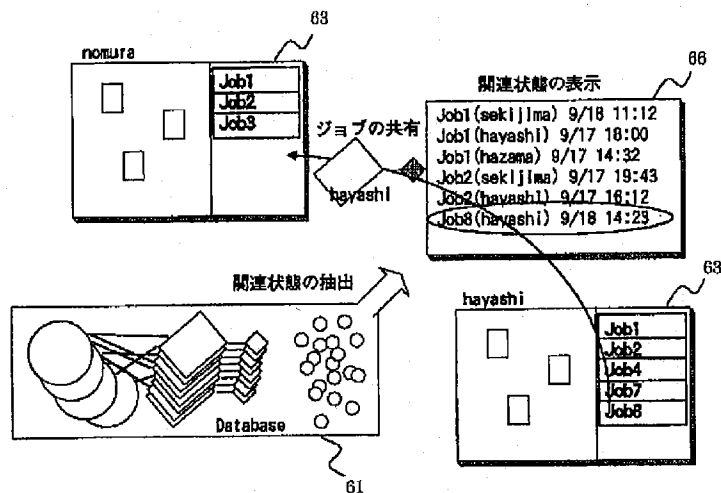
New Cut Copy Paste New Job

Mediator Name	Mediator URL
TR-dyna	TR-dyna
TR-dyna	TR-dyna
TR-dyna	TR-dyna
TR-dyna	TR-dyna

Verifying RSP-Practices Workstory EvsStory Workstory-mk2 CoosS96 Information-Memo PiG test CSCW mass media

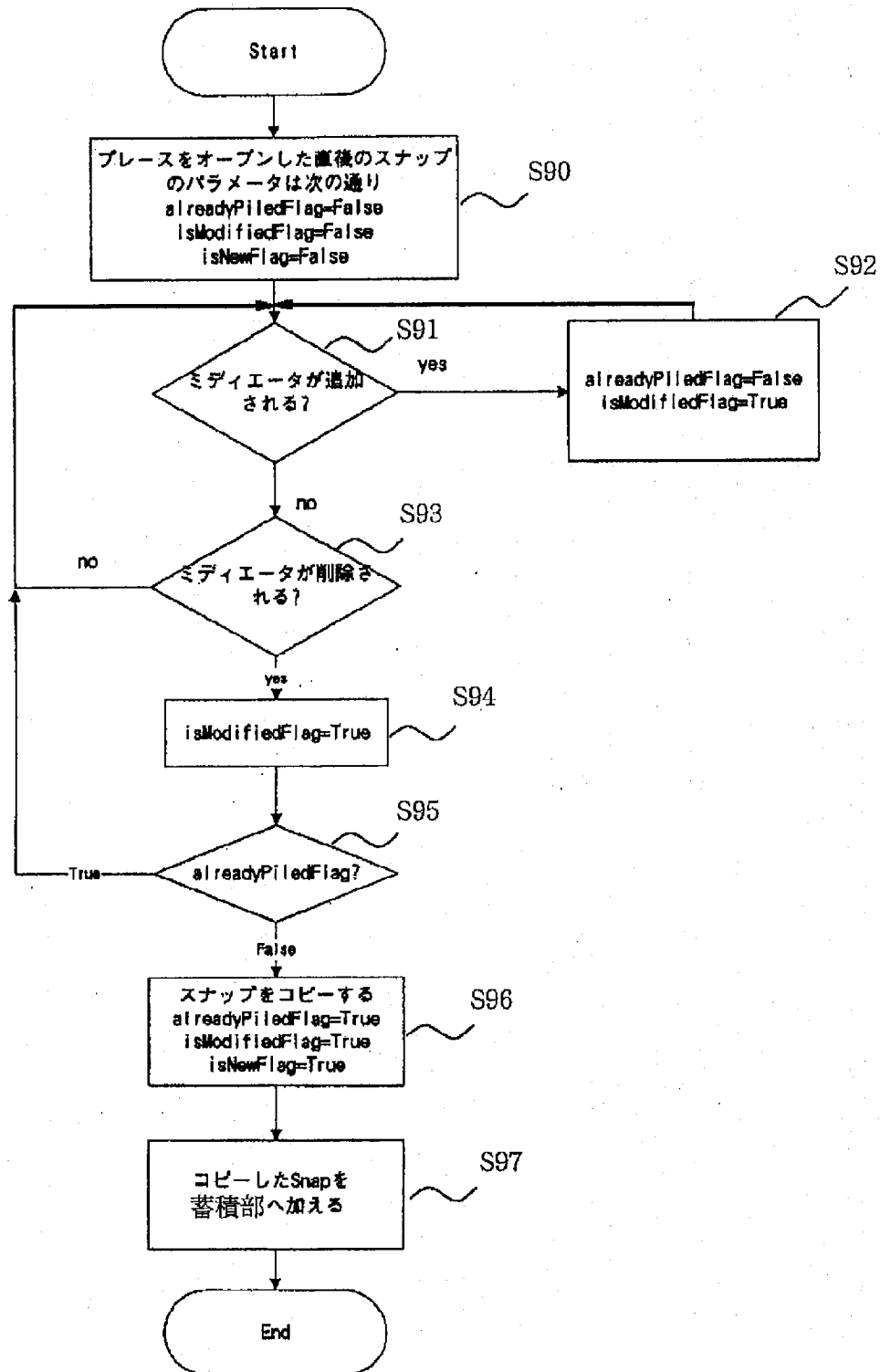
Delete Snap Import Snap

【図52】

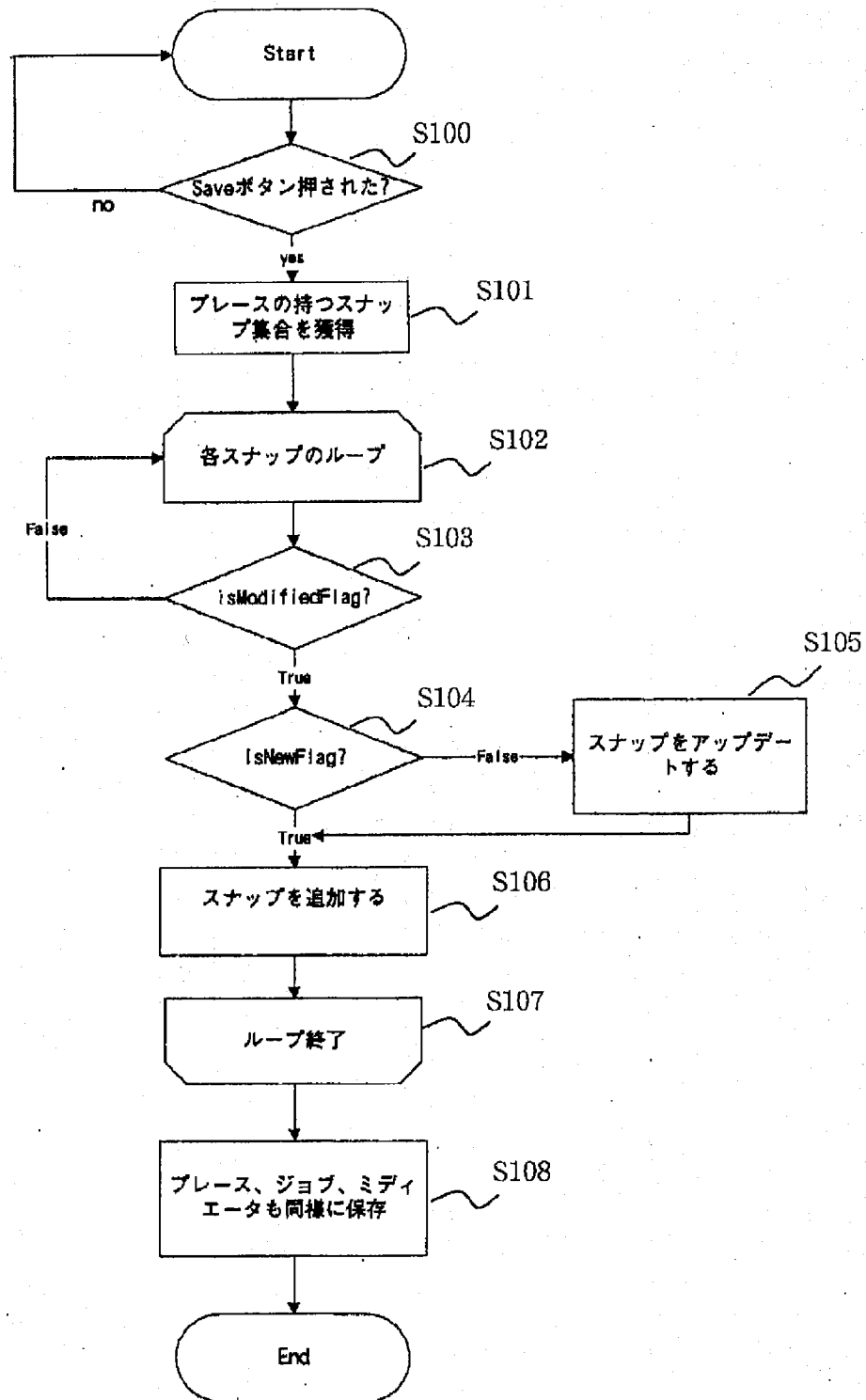




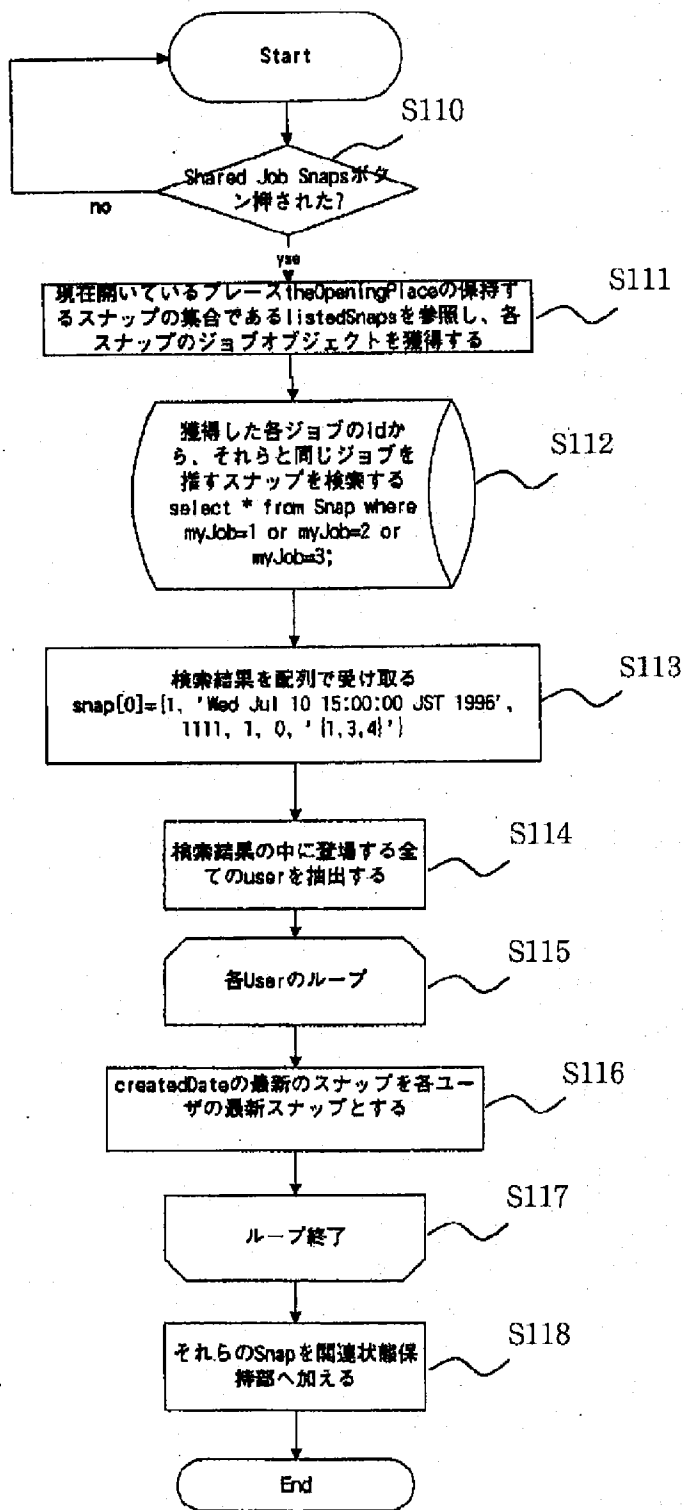
【図46】



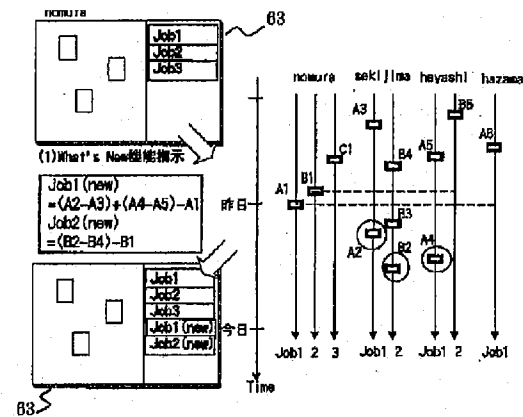
【図47】



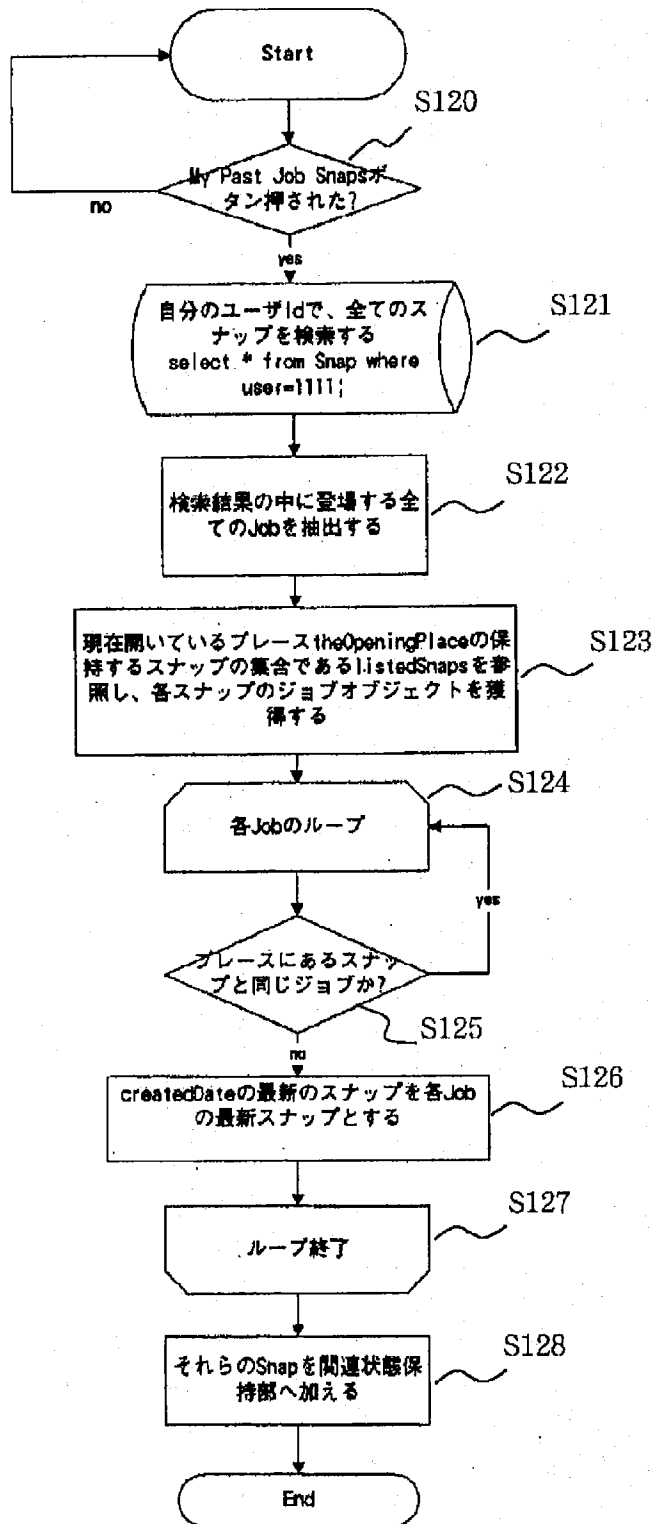
【図49】



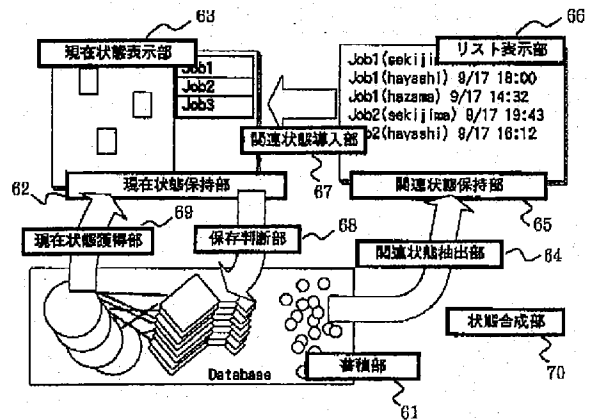
【図62】



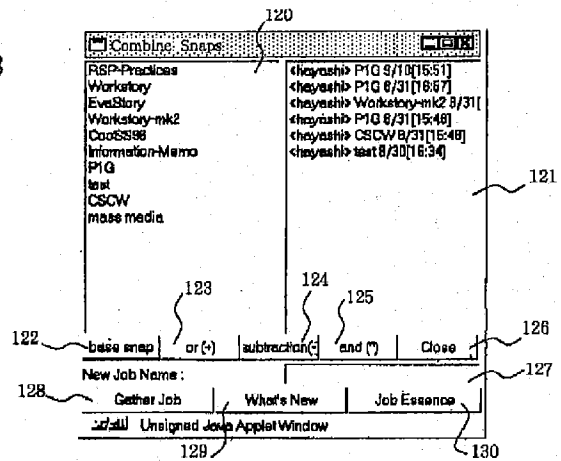
【図51】



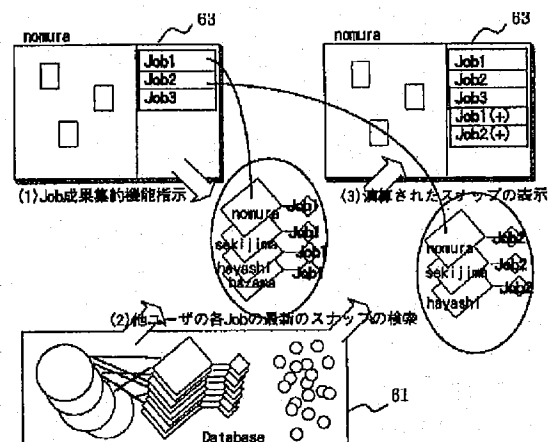
【図58】



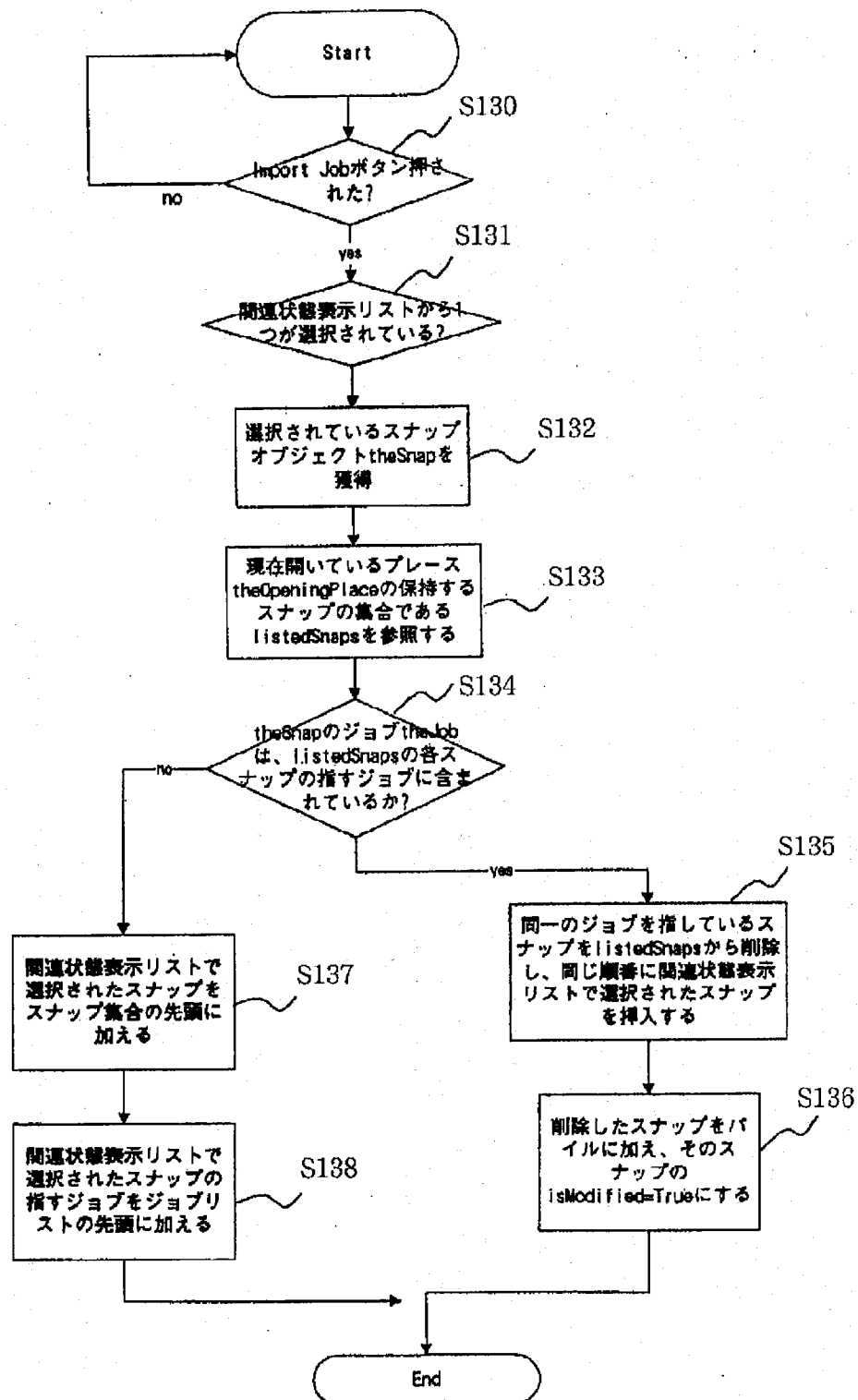
【図59】



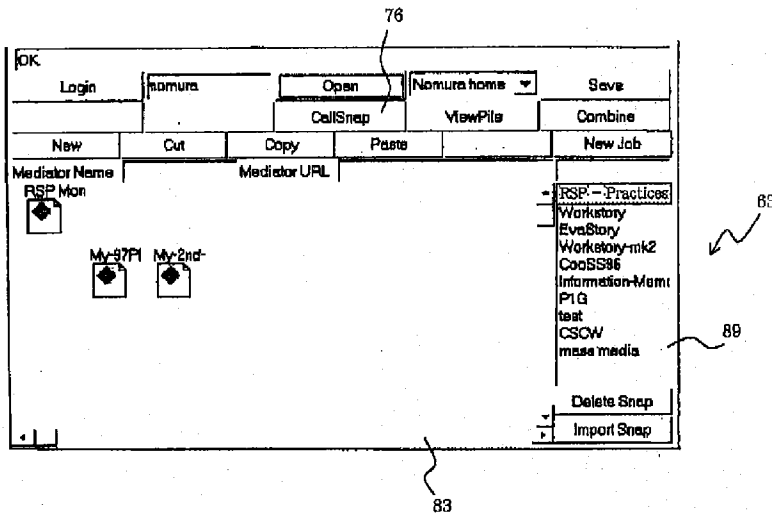
【図60】



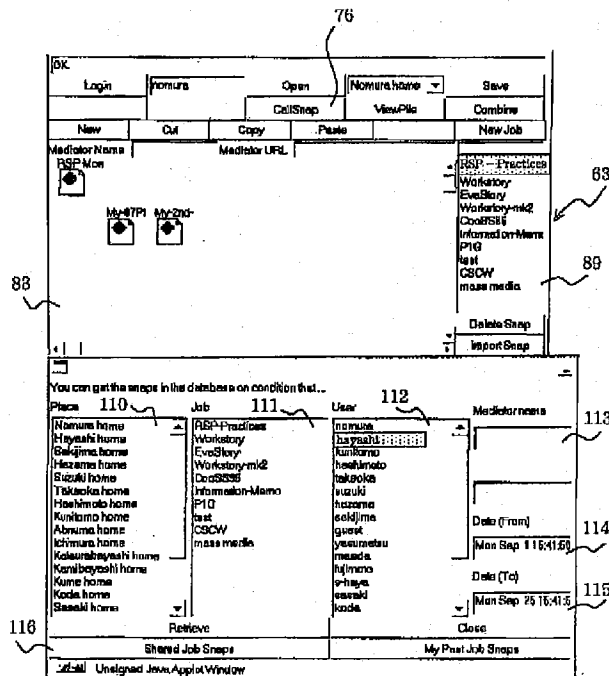
【図53】



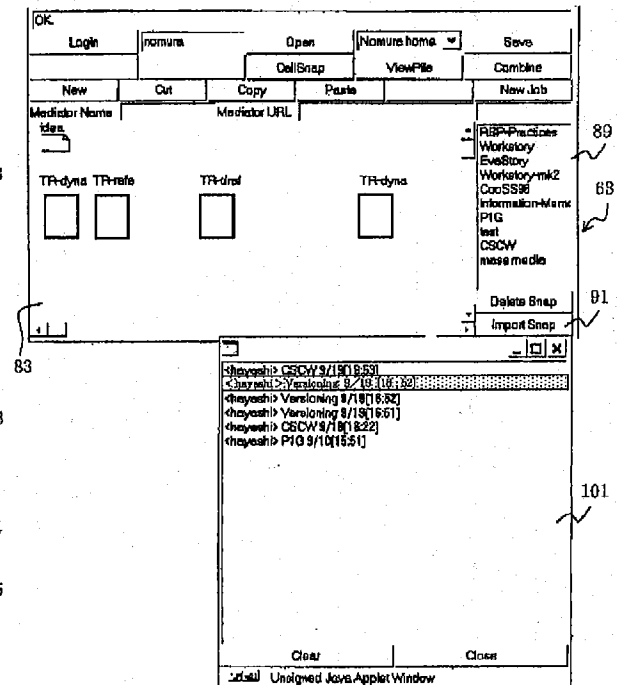
【図54】



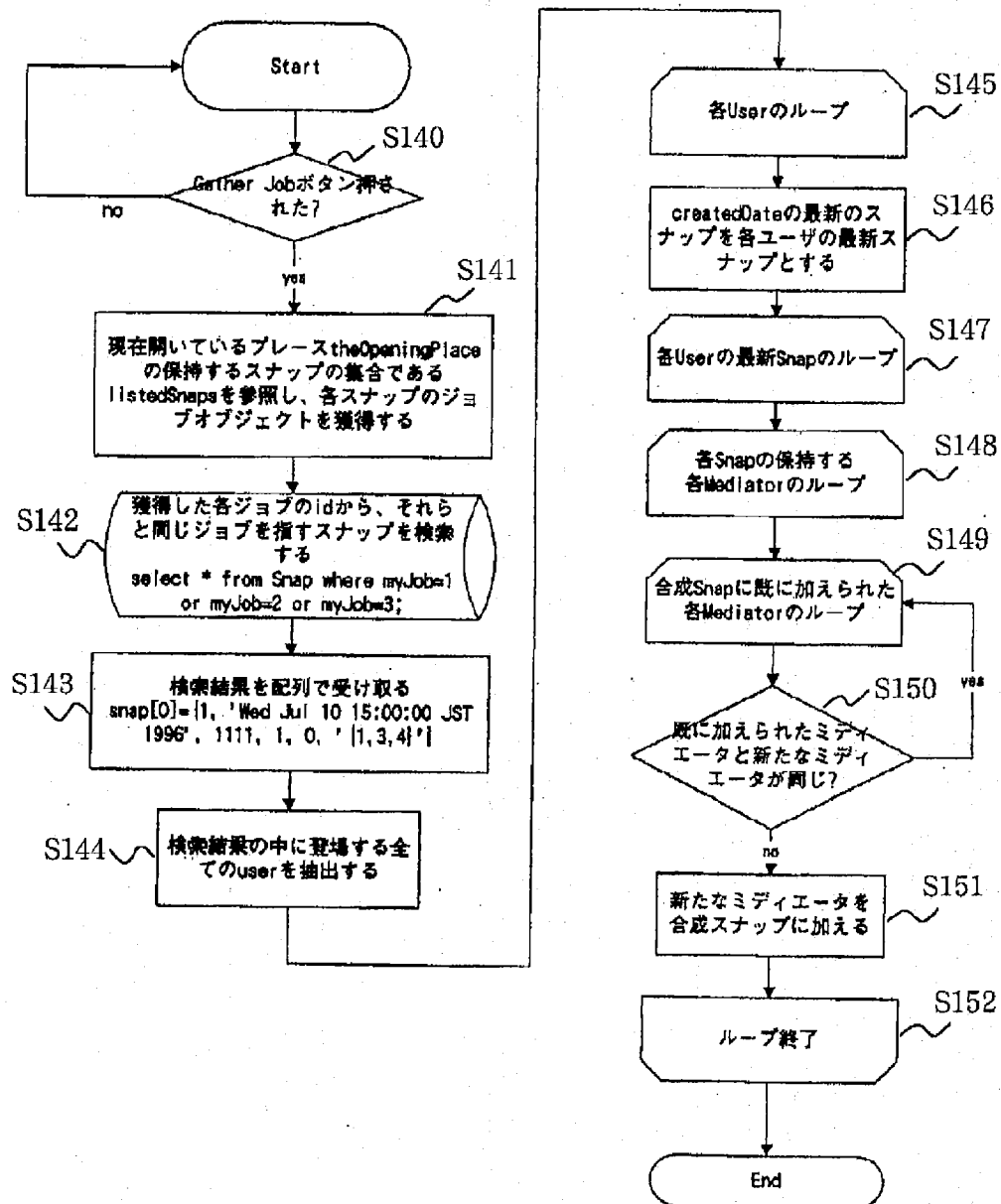
【図55】



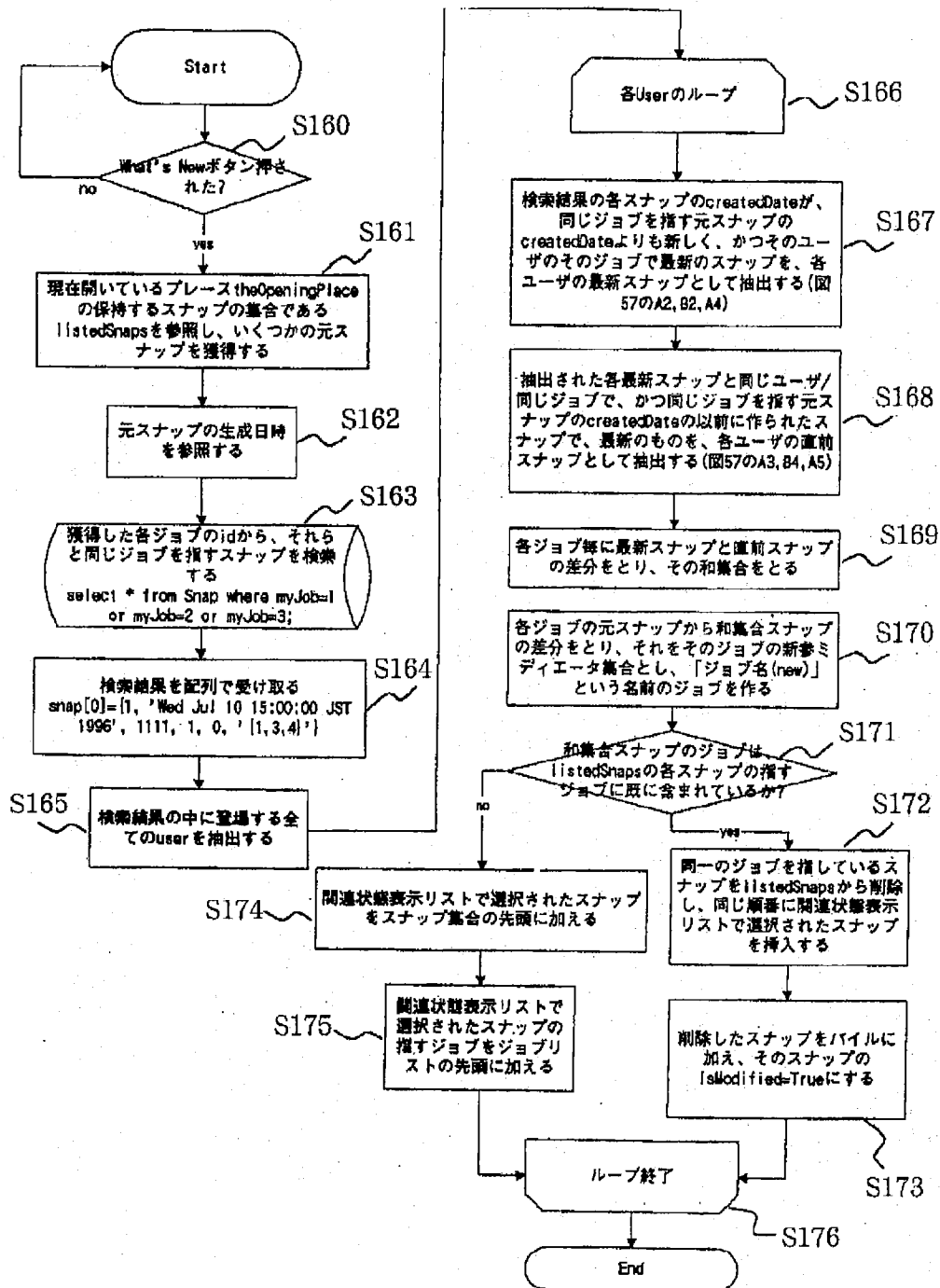
【図56】



【図61】

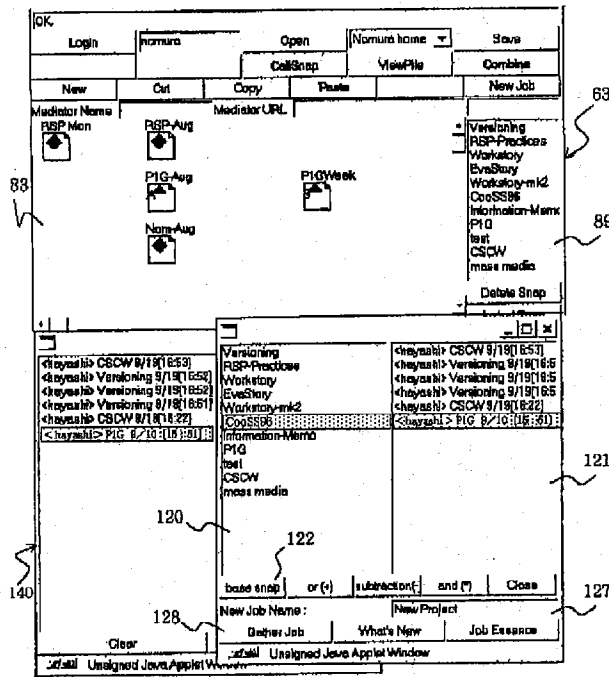


【図63】





【図64】



【図65】

